FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Joint MTU and Rate Adaptive Mechanism for Maritime Wireless Networks

Bruno Rafael Ribeiro Costa

Mestrado Integrado em Engenharia Eletrotécnica e de Computadores

Advisor: Manuel Ricardo (PhD)

Co-Advisor: Rui Campos (PhD)

October 30, 2015

U.PORTO
FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

MIEEC - MESTRADO INTEGRADO EM ENGENHARIA ELETROTÉCNICA E DE COMPUTADORES | **2014/2015**

A Dissertação intitulada

"Joint MTU and Rate Adaptive Mechanism for Maritime Wireless Networks"

foi aprovada em provas realizadas em 16-10-2015

o júri

Presidente Professor Doutor Ricardo Santos Morla
Professor Auxiliar do Departamento de Engenharia Eletrotécnica e de Computadores da Faculdade de Engenharia da Universidade do Porto

Professor Doutor Jorge Botelho da Costa Mamede
Professor Adjunto do Departamento de Engenharia Eletrotécnica Instituto Superior de Engenharia do Porto

Professor Doutor Manuel Alberto Pereira Ricardo
Professor Associado do Departamento de Engenharia Eletrotécnica e de Computadores da Faculdade de Engenharia da Universidade do Porto

O autor declara que a presente dissertação (ou relatório de projeto) é da sua exclusiva autoria e foi escrita sem qualquer apoio externo não explicitamente autorizado. Os resultados, ideias, parágrafos, ou outros extratos tomados de ou inspirados em trabalhos de outros autores, e demais referências bibliográficas usadas, são corretamente citados.

Autor - Bruno Rafael Ribeiro Costa

Faculdade de Engenharia da Universidade do Porto

Abstract

Nowadays, wireless devices are very popular and cost-effective, being the broadband Internet access a big part of our daily life. However, in more remote environments that is not true. A big part of our planet is made up of water, a large unexplored area. Giving access to broadband communications services in maritime range is a trendy subject. Granting Internet access to people while at sea can bring benefits for crew welfare such as keeping in touch with their families and feeling less excluded.

Different technologies are currently available to provide wireless communications at sea, such as WiMAX and VSAT. However, they are not cost-effective for small business or recreational ships, and require complex equipment, which may consume large amounts of power and space. Wi-Fi technology is not currently adapted to maritime environment since channel properties make the degradation and attenuation of the signal a problem for an efficient solution.

The aim of this dissertation is to optimize the MAC layer of IEEE 802.11 for maritime wireless networks, presenting a novel Adaptive Rate and MTU Algorithm (ARMA) that adapts simultaneously the Maximum Transfer Unit (MTU) and data rate taking into consideration the received signal strength.

In order to conduct an experimental test, a prototype was created to validate and compare the behaviour of the algorithm against its state of the art counterparts in a point-to-point communications scenario. The solution presented was tested in the INESC TEC's maritime testbed for this purpose, MARBED and at the Robotics Exercise 2015 (REX15).

The evaluation of the results show that our Adaptive Rate and MTU Algorithm (ARMA) outperforms the most used state of the art rate adaptation algorithm in low signal-to-noise ratio regimes.

Resumo

Hoje em dia os dispositivos sem fios são muito populares e muito baratos e o acesso de alta velocidade à Internet é uma grande parte do nosso dia a dia. No entanto, em locais mais remotos isso não acontece. Grande parte do planeta é constituído por água e está por explorar. Dar acesso a serviços de comunicação de banda larga em ambiente marítimo tem sido alvo de um interesse cada vez maior. Dar acesso às pessoas enquanto estão no mar pode trazer benefícios para o bem-estar das tripulações como por exemplo manter o contacto com a família, o que faz com que se sintam menos sozinhos.

Atualmente estão disponíveis várias tecnologias para garantir acesso a redes de comunicação sem fios no mar, como por exemplo WiMAX e VSAT. Contudo, são demasiado caras para pequenos negócios ou barcos de recreio e requerem equipamento complexo que pode consumir grandes quantidades de energia e ocupar bastante espaço. A tecnologia de Wi-Fi atualmente não está adaptada para ambientes marítimos uma vez que estes têm caraterísticas como a degradação e atenuação do sinal devido à ondulação do mar, que são um problema para as soluções atuais.

O objetivo desta dissertação é adaptar a camada MAC do IEEE 802.11 para comunicações marítimas sem fios apresentando um algoritmo adaptativo do débito e de MTU (ARMA) que adapta simultaneamente o tamanho do Maximum Transport Unit e do débito tendo em consideração o valor de RSSI.

Para realizar os testes experimentais foi criado um protótipo para validar e avaliar o comportamento do algoritmo comparativamente com as soluções homólogas do estado da arte para comunicações ponto a ponto. A solução apresentada foi testada no testbed do INESC TEC para o propósito, MARBED e no Robotics Exercise 2015 (REX15).

A avaliação dos resultados mostra que o algoritmo desenvolvido (ARMA) tem melhor desempenho que o algoritmo de débito adaptativo mais usado, em situações com uma relação sinal-ruído baixa.

Acknowledgments

To all those whom I met somehow, somewhere, sometime and whose demeanor helped me grow as a person, my gratitude.

I would like to thank to my supervisor Prof. Manuel Ricardo for the opportunity granted, allowing me to work and be part of the INESC TEC team. A special thanks to my co-supervisor Rui Campos for the commitment, support and insightful comments provided, Filipe Teixeira and Mário Lopes for the help and overtime during the course of this work.

A word of appreciation to all of those who gave me strength and re-ignited the spark when the light was dim. I would like to express my sincere gratitude to my colleagues whom I came across, walked with me on this path and turned this adventured it in an unforgeable ride. I would like to phrase my thankfulness to my friends, to those always present and to those that even not being with me are always there and just a thought away from making me smile.

A mention to all my family for providing me the conditions to be writing this dissertation right now.

Bruno Costa

"Smooth seas do not make skillful sailors."

Old Proverb

Contents

List of Figures

List of Tables

Abreviantions and Symbols

3G	Third Generation
A-MPDU	MAC Protocol Data Unit Aggregation
A-MSDU	MAC Service Data Unit Aggregation
AARF	Adaptive Automatic Rate Fallback
ACK	Acknowledgment
AMRR	Adaptive Multi-Rate Retry
ARF	Automatic Rate Fallback
ARMA	Adaptive Rate and MTU Algorithm
AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
BO	Back-off
CTS	Clear to Send
DCF	Distributed Coordination Function
DEEC	Departamento de Engenharia Electrotécnica e de Computadores
DIFS	DCF Interframe Space
FER	Frame Error Rate
FEUP	Faculdade de Engenharia da Universidade do Porto
GPRS	General packet radio service
GPS	Global Positioning System
HF	High Frequency
IEEE	Institute of Electrical and Electronics Engineers
IMO	International Maritime Organization
LTE	Long-Term Evolution
MAC	Media Access Control
MCS	Modulation and Coding Schemes
MF	Medium Frequency
MIEEC	Mestrado Integrado em Engenharia Eletrotécnica e de Computadores
MIMO	Multiple Input Multiple Output
MTU	Maximum Transmission Unit
NACK	Negative Acknowledgment
OS	Operative System
RBAR	Receiver Based Auto-Rate
RTA	Real Time Applications
RTS	Request to Send
RTT	Round-Trip Time
RSSI	Received Signal Strength Indicator

SER	Symbol Error Rate
SIFS	Short Interframe Space
SINR	Signal to Interference plus Noise Ratio
SM	Spatial Multiplexing
SNR	Signal-to-noise ratio
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
VHF	Very High Frequency
VLAN	Virtual Local Area Network Terminal
VSAT	Very Small Aperture Terminal
W-CDMA	Wide-Band Code-Division Multiple Access
WiMAX	Worldwide Interoperability for Microwave Access

Chapter 1

Introduction

1.1 Context

Lately, there has been a lot of research, development and investment in wireless communications in maritime environments. Being connected to the Internet has become crucial in our daily life, especially for companies whose business models depend on the connection to the Internet. Making presence on the grid is essential for every company. However, small businesses related to the sea, such as traditional fisheries, do not have access to the Internet while at sea. To overcome this limitation, current technology, such as satellite communication is expensive and requires specialized hardware and dedicated antennas [1, 2], that small watercraft do not have. There are many other examples for the use of communications services at the sea like cruise ships or even buoys retrieving data.

Most of systems use more than one technology combined. Having a cost-effective, reliable solution for wireless communication in maritime environments brings large benefits. Wi-Fi, or IEEE 802.11 is nowadays a very affordable technology and can be found in most mobile communications devices. From now on, the terms IEEE 802.11 and Wi-Fi are used interchangeably. Wi-Fi still holds some limitations, for instance the distance or delay, but is a progress for wireless maritime communications. We can extend the communication range using mesh networks, although this has repercussions at the delay level. There is some promising research [3] to try to make this standard more suitable for this scenario, adding new functionalities to this widely used and low-cost technology making it a feasible option and to bring new development to this field of study.

1.2 Motivation

Having an affordable solution for wireless communications can bring seafarers closer to their loved ones while at work. Even tourism ships or private boats could use broadband access to the Internet and more applications could be explored if Real Time Applications (RTA) were feasible. So, adapting the existing mechanisms to the rough conditions of the sea context is the step to take.

The 802.11 MAC standard is not ideal for a communications channel with a mutable quality. The selected transmission rate for a given moment is the ideal one for moments before, when the measurements of the channel quality were taken. So it uses the right algorithm for some time in the past.

At the sea, channel conditions are constantly changing. The boat moves with sea undulation, producing an unstable link for communications . Using models that describe wave movement may be a promising approach to create an algorithm that optimizes the rate and MTU for maritime environments, predicting the suitable parameters to obtain the highest possible throughput.

1.3 Objectives

The main goal of this thesis is to improve the Wi-Fi MAC in maritime wireless communications, as seen in Figure 1.1, by creating a suitable rate adaptive algorithm and a dynamic variation of the MTU to enhance the communication.

Figure 1.1: Communication between shore-to-ship.

In order to achieve this goal, we aimed to:

• Identify and characterize the relation between the water oscillation and the communication channel conditions;

• Analyse prior experiments done with Wi-Fi 802.11 at 5.8 GHz [4];

• Create a solution to incorporate a new method using dynamic variation of the MTU and a rate adaptive mechanism to improve the communication;

• Implement the algorithm in Linux/OpenWRT;

• Create a communications prototype in order to test the solution at Robotics Exercise 2015;

• Test the new optimization system under controlled conditions and in the INESC TEC's testbed for the purpose;

• Evaluate the proposed solution and compare it with its state of the art counterparts.

1.4 Structure

This document is composed of 6 chapters. Chapter 2 describes the state of the art and pertinent work is described. Chapter 3 presents the leading thread and an explanation of Adaptive Rate and MTU Algorithm (ARMA). Chapter 4 explains the methodology used in the implementation of ARMA as well the major hardware and software components. It also clarifies some of the auxiliary tasks for the data analysis. Chapter 5 describes the different test scenarios and discusses the achieved results. Finally, Chapter 6 draws the conclusions about all the work done and and points out future work.

Chapter 2

State of The Art

This chapter provides some underlying knowledge for a better understanding of the proposed approach. We show some maritime communications systems already implemented. An introduction to MTU is made and some of the most important rate algorithms for the 802.11 that are mentioned. We also provide some representation of radio propagation models as well as models for the waves at the sea surface.

2.1 Maritime Communications

In this section we introduce some of the options already used for maritime communications. We can see in Table 2.1 we have a few examples of the bandwidth used for each technology [5]. Some of these systems, like TRITON, are mesh networks with ongoing research and development. Also, in Table 2.2, some current maritime networks and systems and the technology that they use are presented, comparing their range, where ++ is a higher maximum range and - lower. A brief introduction is made to IEEE 802.11 evolution. This is the technology that will be used in the final implementation.

Table 2.1: Channel bandwidth for different technologies.

Technology	**Bandwidth**
MF/HF	0.5 - 3 kHz
VHF	25 kHz
WiMAX	10 MHz
LTE	1.4 - 20 MHz
LTE - Advanced	up to 100 MHz
IEEE 802.16m	5 - 20 MHz

Table 2.2: Qualitative analysis of maritime network technologies ranges (adapted from [6]).

System/Network	Technology	System/Network Maximum Range
AIS	VHF	+
PACTOR-III	MF/HF	++
GMDSS	HF, HF, VHF COSPAT-SARSAT	++
NANET	WiMAX, LTE IEEE 802.16m MF/HF, VHF	+
TRITON	IEEE 802.16e, WiMAX	-
IWMCN	IEEE 802.11s, INMARSAT FB W-CDMA	-
MaritimeManet	VHF, IEEE 802.11	-
MarCom	LTE, WiMAX, D-VHF IEEE 802.11, IEEE 802.16	+

2.1.1 Very-Small-Aperture Terminal (VSAT)

There are already some implementations on maritime communications using satellites, as illustrated in Figure 2.1. VSAT is used in long range satellite communications.

Figure 2.1: Example of a maritime communication via satellite .

Due to the sea conditions it is very difficult do keep the beam and the antenna aligned and highly precise stabilized antenna should be used [2, 7]. The connection speed on such devices is around 1 Mbps. Normally VSAT communication is made through geostationary orbit satellites that can use Ku-band, C-band or Ka-band. Nevertheless, satellite communications still remain too expensive for small maritime commercial services.

One example of this system is Inmarsat[1]. A very well known company created by the International Maritime Organization (IMO) that nowadays has a fleet composed by several satellites that can provide communication services at the sea [8].

2.1.2 WiMAX

WiMAX is a wireless technology that implements IEEE 802.16. Latest implementations include three main components: base station (BS), mobile station (MS) and relay station (RS). The communication is made from BS to MS or vice-versa and the RS is used as intermediary to enhance the range and link quality. Some other relevant experiments in the field like WISEPORT or TRITON use this technology in their projects. WiMAX has two main components: WiMMAX fixed and WiMAX mobile. The evolution of this standard allows the frequency range of 10 to 66 GHz and some from 2 to 11 GHz too [9]. WiMAX typically requires licensed bands, which carries on additional costs when deploying this technology [10].

2.2 IEEE 802.11a

IEEE developed a protocol specification for wireless local area network (WLAN) that defines the wireless interface, oriented to the MAC and physical layers, for different nodes to communicate. At the moment, Wi-Fi 802.11 is widely used and the fact that 802.11 uses a licence-free spectral space is one of the reasons.

One of the earlier extensions of the IEEE 802.11 specifications was IEEE 802.11a. The *a* amendment uses orthogonal frequency division multiplexing (OFDM) as encoding method. It operates on 5 GHz band and 20 MHz channel bandwidth. Table 2.3 shows the modulation type, coding rate and data rate used.

Table 2.3: 802.11a Data rates, modulation types and conding rates.

Data Rate (Mbit/s)	Modulation	Coding Rate
6	BPSK	1/2
9	BPSK	3/4
12	QPSK	1/2
18	QPSK	3/4
24	16-QAM	1/2
36	16-QAM	3/4
48	64-QAM	1/2
54	64-QAM	3/4

The control of the shaded physical medium is an important part of the IEEE 802.11 standards. The Distributed Coordination Function (DCF), an example in Figure 2.2, is a primary Medium

[1] http://www.inmarsat.com/

Access Control (MAC) procedure based on CSMA/CA. If the medium is not busy for a period of time (DIFS), after a Back-off (BO) time (Equation 2.2), the data is sent. This BO time is a multiple of the time of one slot, and is used to prevent that, if multiple stations find the channel released, transmit at the same time and so avoiding collisions. We wait the SIFS interval for a ACK response. And we start over the cycle to send data again.

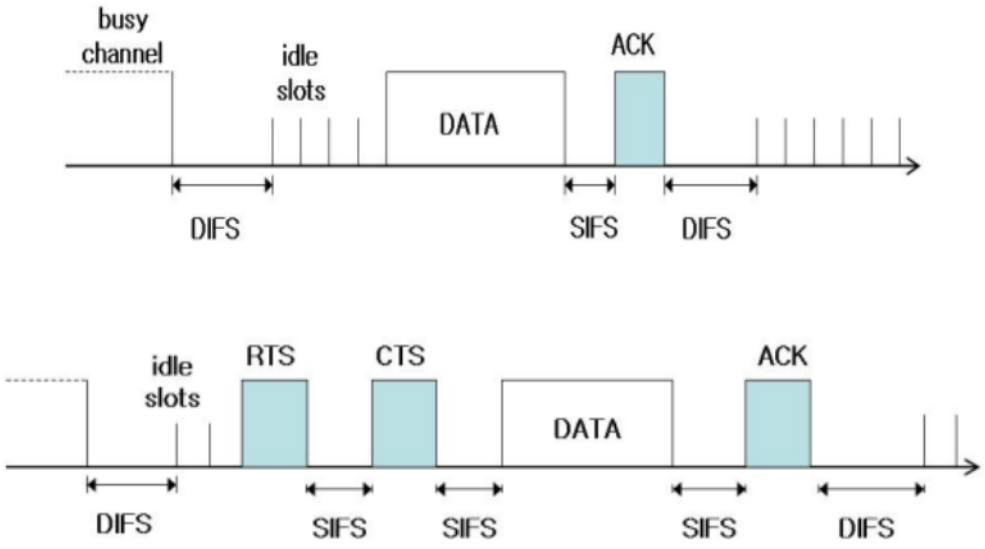


Figure 2.2: IEEE 802.11g/a DCF Channel access mechanism (basic access and with RST/CTS) [11].

DCF Interframe Space (DIFS, Equation 2.2) time can be calculated as a function of Short Interframe Space (SIFS) time that is specific for each standard[2]. For IEEE 802.11a, a time slot lasts 9 μs, SIFS time for 5 Ghz is equal to 16 μs and so DIFS 34 μs.

$$DIFS = SIFS + (2 \times SlotTime) \tag{2.1}$$

$$\text{Backoff\_Time} = \text{random}() \times \text{SlotTime} \tag{2.2}$$

More deconvolved mechanisms were created to reduce frame collisions caused by problems like hidden nodes. The optimal resolution for this problem is DCF with Request to Send / Clear to Send frames as in Figure 2.2. Introducing additions RTS/CTS method can prevent the hidden node problem and help to prevent collisions. However, RTS/CTS control frames also create overhead on the communication.

[2]http://standards.ieee.org/getieee802/download/802.11-2012.pdf

2.3 IEEE 802.11n

Due to the increasing hunger for more information and faster, the IEEE 802.11 standard evolved and IEEE 802.11n is one of the most used nowadays. IEEE 802.11n implements additional functionalities to improve, for example, bitrate and reduce propagation loss effects, etc. The norm has been evolving and 802.11n can get theoretically rates until 600 Mbps.

There were already some work done adapting Wi-Fi to maritime conditions. For example in [12] ship-to-shore and shore-to-ship communications up to 1 Mbit/s until 9 km distance are shown. In 802.11n new concepts were introduced to improve the connection compared to its predecessors. Some of these features influence the throughput tested in long range communications [13].

2.3.1 Multiple Input Multiple Output (MIMO)

MIMO technology allows the use of several antennas to send and/or receive data simultaneously. For example, on one side, two antenna and on the other three (Figure 2.3). In IEEE 802.11n diverse MIMO configurations are specified that goes from one to one antenna, until four to four.

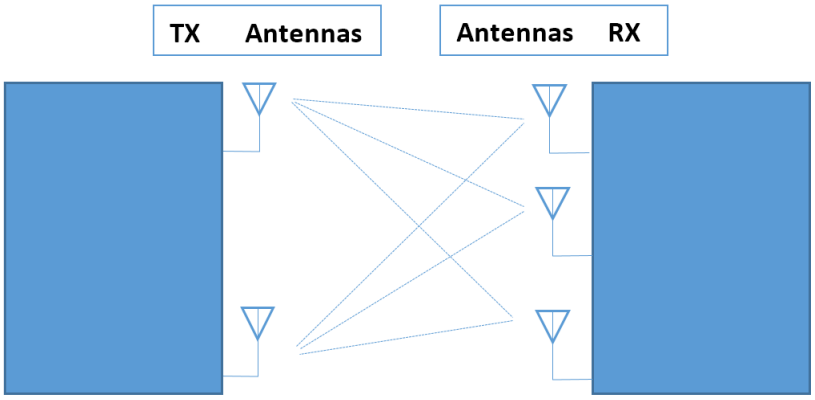


Figure 2.3: MIMO system with multiples antennas.

Advanced Spatial Multiplexing (SM) was also developed to follow this evolution [14]. Using MIMO in maritime communications is expected to increase bitrate and help to overcome channel issues like wave shadowing and reflection.

2.3.2 40 MHz Channel Bandwidth

IEEE 802.11n can also use a 40MHz channel instead of the previous 20 MHz from IEEE 802.11 a/b/g networks. Working a channel with 40 MHz will increases throughput as more bandwidth is used. Nevertheless, it is more susceptible to interference. In addition, considering the same conditions, doubling the bandwidth will also double noise power.

Wider channels should be used when the channel quality is very good and the receiver is close to the transmitter to boost the throughput but we can still use narrow ones when the interference is a major factor [15].

2.3.3 Modulation and Coding Schemes (MCS)

The IEEE 802.11n standard defines different MCS, as seen in Table 2.4. This combinations are selected between devices. Different devices and channel conditions will result in different type of modulation and coding selected. To know what rate is being used we need to know, besides MCS index, the bandwidth used and guard interval (generally in *ns*).

Table 2.4: Some 802.11n MCS examples [15].

MCS Index	Type	Coding Rate	Spatial Streams	Data Rate (Mbps) with 20 MHz CH		Data Rate (Mbps) with 40 MHz CH	
				800 ns	400 ns (SGI)	800 ns	400 ns (SGI)
0	BPSK	1 / 2	1	6.50	7.20	13.50	15.00
1	QPSK	1 / 2	1	13.00	14.40	27.00	30.00
2	QPSK	3 / 4	1	19.50	21.70	40.50	45.00
3	16-QAM	1 / 2	1	26.00	28.90	54.00	60.00
4	16-QAM	3 / 4	1	39.00	43.30	81.00	90.00
5	64-QAM	2 / 3	1	52.00	57.80	108.00	120.00
6	64-QAM	3 / 4	1	58.50	65.00	121.50	135.00
7	64-QAM	5 / 6	1	65.00	72.20	135.00	150.00
8	BPSK	1 / 2	2	13.00	14.40	27.00	30.00
9	QPSK	1 / 2	2	26.00	28.90	54.00	60.00
10	QPSK	3 / 4	2	39.00	43.30	81.00	90.00
11	16-QAM	1 / 2	2	52.00	57.80	108.00	120.00
12	16-QAM	3 / 4	2	78.00	86.70	162.00	180.00
13	64-QAM	2 / 3	2	104.00	115.60	216.00	240.00
14	64-QAM	3 / 4	2	117.00	130.00	243.00	270.00
15	64-QAM	5 / 6	2	130.00	144.40	270.00	300.00
16	BPSK	1 / 2	3	19.50	21.70	40.50	45.00
...	...	...	...	...	...	...	...
31	64-QAM	5 / 6	4	260.00	288.90	540.00	600.00

2.3.4 Overhead Reduction

802.11n also brings enhancements, such as block ACK and frame aggregation, to reduce the unnecessary overhead in some frames. Instead of sending one ACK for each frame each time, we can chose to gather several ACK in one block of ACK's to confirm several frames. Using this mechanism instead of sending one ACK for each frame reduces unproductive traffic caused by ACK's.

Frame aggregation is intended to augment the efficiency by fusing the payload into only one transmission, reducing significantly the replicated overhead. Different aggregation methods exist, such as MAC Service Data Unit Aggregation (A-MSDU) or MAC Protocol Data Unit Aggregation (A-MPDU in Figure 2.4). Both methods can be used simultaneously but operate at different levels. We call MSDU aggregation when several payload have equivalent properties (all the payloads must be sent from the same transmitter to the same receiver) and a higher layer concatenates them into a MPDU. The idea behind MPDU aggregation is to append several MPDU with the same PHY header, but it can only be applied after the MAC encapsulation process.

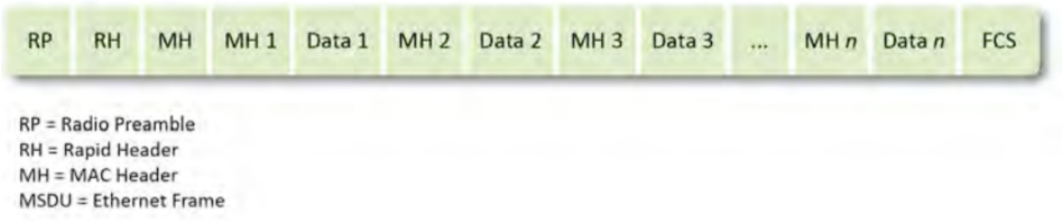


Figure 2.4: MPDU Frame aggregation example [16].

2.4 MTU Variation

Under normal circumstances there is a convention that the most suitable MTU is 1500 bytes (used in other technologies like Ethernet frames), being the default value in most devices. However, there are some studies that support that varying the size of the MTU have better performance in different circumstances [17, 18].

There are also some attempts to create a dynamic algorithm in order to improve Wi-Fi MAC maritime communications performance. In [3] a dynamic algorithm is proposed that adapts MTU size accordingly with RSSI values. Therefore, the author in [3] estimates packet loss based on the RSSI value received at on moment. A prediction algorithm is used to foretell the RSSI value for the next moment in time. This predicted value is used to adjust the MTU size to the correspondingly threshold value.

The prediction algorithm follows a sinusoidal step wave (Figure 2.5) with 8 different positions. The real RSSI values are read and feed into a vector. The predicted RSSI values are updated whenever a new real RSSI value is obtained. Between reads the position on the wave, the time until the next prediction based on the wave period, and the corresponding RSSI predicted value are calculated.

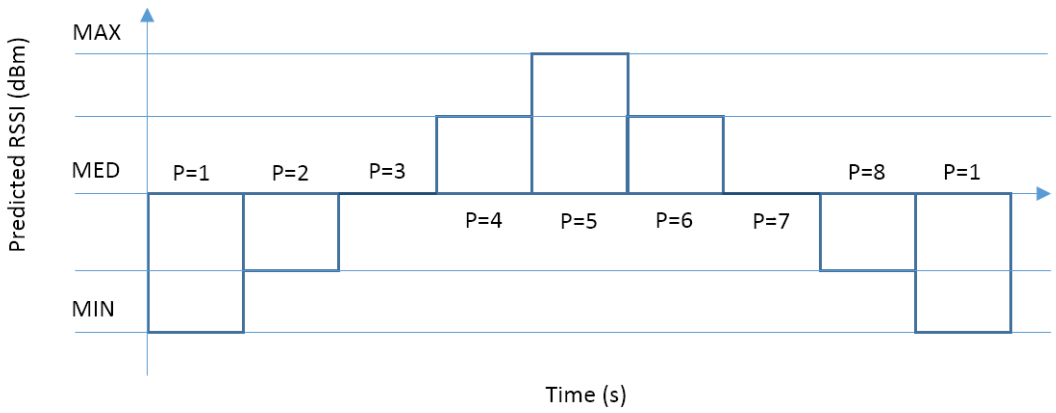


Figure 2.5: Sinusoidal step wave model.

This RSSI prediction algorithm combined with a Dynamic MTU algorithm showed specially beneficial in weak Wi-Fi links and exhibited enhancements to the quality of the communication up to a maximum gain of 133% for TCP traffic.

2.5 Rate Adaptive Algorithms

Rate adaptive algorithms are control mechanisms implemented to maximize throughput or other hardware related parameters, for example power consumption. They resort to a group of constraints to assess the channel state and settle an appropriate data rate.

There are two major groups based on different ways to select the most appropriated rate. To ascertain channel state information obtained from the physical layer or the link layer can be used. From physical layer values like SNR, or equivalents, can be used to represent physical characteristics of the channel and thus decide the best rate to use. The other group, uses link layer information of the channel that can be extrapolated from the retransmission rate or packet loss values [15].

Table 2.5: Different rate adaptation schemes (adapted from [19]).

Algorithm	**Decision based on**
ARF	Loss ratio
AARF	Loss ratio
Onoe	Loss ratio
SampleRate	Loss ratio
Minstrel	Loss ratio
RBAR	SNR
OAR	SNR

2.5.1 Physical Information Based

Most of this type of algorithms uses physical layer information or exchanges the information between the sender and the receiver. The knowledge of SNR or RSSI, for example, is analysed for the data rate selection. This option is usually more responsive because it uses direct measures from the channel. However, it relies on a wireless card interface that only provides limited data.

2.5.1.1 Receiver Based Auto-Rate (RBAR)

RBAR recovers information from the channel through physical layer, and use it to select the bitrate adoption. This measurement is done at the receiver so this choice is made by the its perception of channel conditions and has to pass to the sender, which uses bandwidth and limits the performance. Using RTS/CTS and ACK or NACK, the data rate and transmission power are adjusted [20]. This algorithm is more of a base to other rate adaptation algorithms because, in practical terms, it implies changes to the standard implementation. So one big problem is the incompatibility with IEEE 802.11.

2.5.1.2 Opportunistic Auto Rate (OAR)

The OAR algorithm uses the same probing mechanism as RBAR. The difference is how it transmits the data frames. After the data rate selection by the source node, depending on the data rate selected, OAR sends several consecutive frames. OAR uses IEEE 802.11 fragmentation in its favour in order to have a fair time management and so increasing the throughput.

2.5.2 Loss Ratio Based

The link layer based algorithms deduce the quality of the channel based on the percentage of frames transited with success or the loss rate in a determined period of time. Some of them even make use of RTS/CTS information which imply additional overhead. Some examples are Automatic Rate Fallback (ARF) and some derivatives like Adaptive ARF (AARF) and Adaptive Multi-Rate Retry (AMRR), Onoe, SampleRate or Minstrel. Minstrel is one of the most used algorithms, made to maximize IEEE 802.11 throughput and will be the one used evaluate our algorithm performance.

2.5.2.1 Minstrel

This is one of the most commonly used algorithms for rate control and the default for some Operative Systems (OS), mainly because of its efficiency and availability in wireless drivers. It uses information gathered from the packet trades to assess the channel quality. The Minstrel algorithm is composed by three main parts: 1) a multi-rate retry chain, 2) the rate selection decision and 3) calculation of new statistics. The multi-rate retry chain mechanism is composed by four pares $r0/c0$, $r1/c1$, $r2/c2$, and $r3/c3$ respectively. It starts to try to send a packet at $r0$ rate $c0$ times. If the frame is not transmitted, the algorithm selects another rate $r1$ and try to send $c1$ times and

so on. At the end of all *cn* attempts, if not successful, the frame is discarded. First rate ($r0$) is the one predicted to allow the highest throughput. One feature that differentiates Minstrel from other algorithms alike is how the rate selection is made.

Minstrel uses information gathered from the packet trades to assess the channel quality. Around 10% of the frames are used as samples to estimate success rate of transitions. In the Table 2.6 is shown an enhanced version of the rate selection used by Minstrel with the information from either normal frames or sample frames with rate information.

Table 2.6: Minstrel rate selection [21].

Rate	**Sampling Transmission**		**Normal Transmission**
r_i	**Random <Best**	**Random >Best**	
r_0	Best throughput	Random rate	Best throughput
r_1	Random rate	Best throughput	Next best throughput
r_2	Best probability	Best probability	Best probability
r_3	Lowest Baserate	Lowest baserate	Lowest baserate

Using a probabilistic model Minstrel calculates the success percentage and expected throughput for each rate. This procedure is done regularly based on statistics history [22, 23].

2.5.2.2 The Adaptive Automatic Rate Fallback (ARF)

The AARF algorithm is based on the ARF algorithm. AARF was an evolution for more stable environments. The transmission starts at the lowest available data rate. After a set time interval or N successful transmissions the data rate is increased and reset the time interval. If the first transmission after a data rate change fail, data rate is set for the one before and N threshold is set to $2N$. With this adaptive threshold, in stable channel, it will try to increase the data rate less times than the prior ARF where N was constant. In the Figure 2.6 we have an example of a stable system with data rate of 9 Mbit/s and the comparison between the number of attempts from ARF and AARF. We can see, for this example, where N is 1 second, as time passes, the AARF has less attempts to change the data rate.

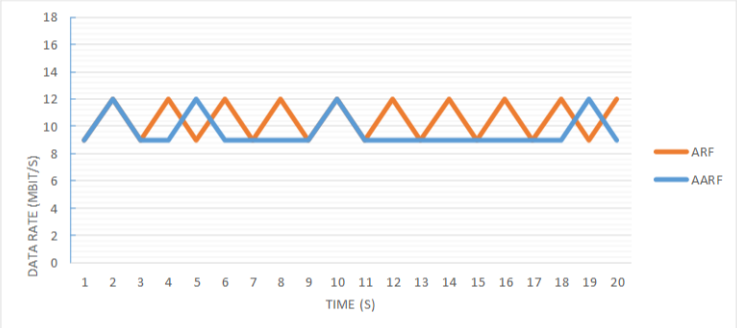


Figure 2.6: Comparison between ARF and AARF.

2.5.2.3 Onoe

Onoe was one of the first rate adaptation algorithm to be implemented on a Linux 802.11 wireless driver. Onoe uses a credit classification system. It seeks the data rate with the most credits, having a loss ratio less than 50%. In IEEE 802.11a/g it starts with data rate of 24 MBit/s, with 0 credits. Every 1 second it does a new assessment, considering the transmission results. If the credit score is 10 or more it switched to the next highest data rate. The credit count is augmented if the retry ratio is less than 10%, if not, the credit total is decreased. If the retry ratio is bigger than 1, the data rate is decreased.

2.6 Bit Error Rate (BER) and Signal-to-Noise Ratio (SNR)

Bit error ratio and throughput theoretical variation with signal-to-noise ratio are presented in Figure 2.7 considering an AWGN channel, assuming a symbol rate of 1 mega-symbol per second and a packet size of 1500 bytes[22]. As seen, as SNR increases, the BER decreases. For smaller BER we can use different modulations with lower BER and higher throughput. In maritime ambient if the channel quality, and so the SNR, fluctuates the optimal coding will also vary.

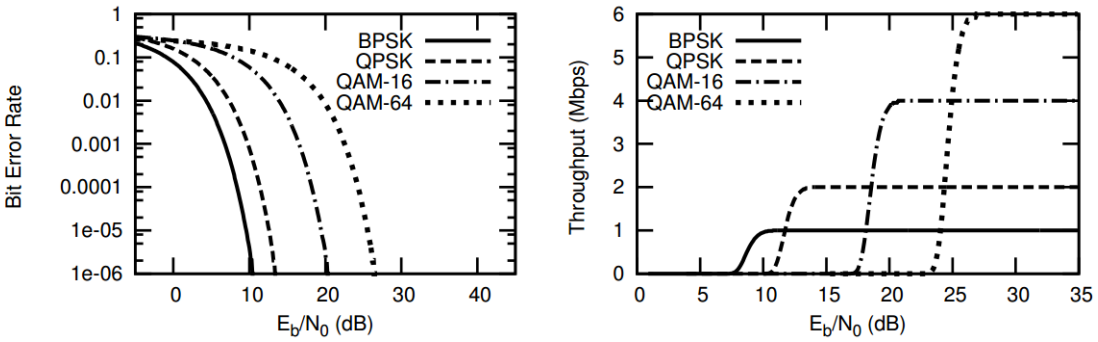


Figure 2.7: Comparison Between BER and SNR values and throughput and SNR values [22] .

IEEE 802.11 standard considers the possibility of using different communication rates according to the changes on the channel quality. Them maximum data rate possible [24] is given by

$$Maximum\ data\ rate\ = \frac{Data\ carriers\ x\ Spacial\ Streams\ x\ Bits\ per\ Symbol\ x\ Code\ Rate}{Symbol\ Duration} \tag{2.3}$$

In IEEE 802.11n the maximum value for data carriers is 108, spacial streams is 4, bits per symbol is 6, code rate is 5/6 and Symbol Duration is 3.6 μs giving us

$$Maximum\ data\ rate\ = \frac{108\ x\ 4\ x\ 6\ x\ 5/6}{3.6\ \mu s}\ =\ 600\ Mbps \tag{2.4}$$

In case of channel degradation, the bitrate is reduced to maintain the BER. BER is very important to the channel quality decision. It has some relation with SNR and respectively modulation used. In [25] the author refers to a representation for wireless modulation performance. It discusses the relation between BER and several different modulations for AWGN channels. Considering the SNR value, where B is the signal Bandwidth and T_b is the bit time:

$$\text{SNR} = \frac{E_b}{N_0} \times \frac{1}{B \cdot T_b} \tag{2.5}$$

In a wireless channel error can occur due to collisions or transmission errors. Collisions may be caused by different nodes creating traffic simultaneously. Errors in transmissions can occur due to various causes like, for example, a fragile link and loss of connection.

Packet loss probability (P_L) is defined in order of collision probability (P_c) and error probability (P_e) :

$$P_L = 1 - (1 - P_c)(1 - P_e) \tag{2.6}$$

Considering a packet, the aggregation of the header and payload, where the header with the length L_h is transmitted at the rate R_h and the payload with the length L transmitted at a rate of R_p, the probability of an error occur is :

$$P_e = 1 - (1 - BER_{R_h})^{L_h} (1 - BER_{R_p})^{L} \tag{2.7}$$

The authors in [26] and [27] consider for the Symbol Error Rate (SER) in an uncoded AWGN channel for the following modulations:

- **BPSK**

$$SER = Q\left(\sqrt{2\frac{E_b}{N_0}}\right) \tag{2.8}$$

- **QPSK**

$$SER = 2 \cdot Q\left(\sqrt{2\frac{E_b}{N_0}}\right)[1 - \frac{1}{2} \cdot Q\left(\sqrt{2\frac{E_b}{N_0}}\right)] \tag{2.9}$$

- **M-QAM**

$$SER = 4 \cdot \frac{\sqrt{M}-1}{\sqrt{M}} \cdot Q\left(\sqrt{\frac{3}{M-1}\frac{\log_2 ME_b}{N_0}}\right) - 4 \cdot \left(\frac{\sqrt{M}-1}{\sqrt{M}}\right)^2 \cdot Q^2\left(\sqrt{\frac{3}{M-1}\frac{\log_2 ME_b}{N_0}}\right) \tag{2.10}$$

This equations are used in a MATLAB[3] module to compute bit and symbol error rates. Considering M as the number of points in the constellation, the number of bits per symbol are $k = \log_2 M$, and we can use an approximation of the Bit Error Rate (BER) considering:

$$BER \approx \frac{SER}{\log_2(M)} \tag{2.11}$$

Note that increasing the bitrate when good channel conditions exist increases the efficiency of the communication. So this balance is very important for the performance of the communication. Also it must agree with the modulation scheme used because higher modulation schemes encode more data per time unit.

The wireless channel is constantly changing, even more in maritime ambient where the conditions are rough. Having a method to dynamically choose the better rate to use is needed. In order to do this, two main problems must be faced: channel quality and estimation rate selection.

Channel quality evaluation was discuses above. For the rate selection a common technique used is the comparison with a table of values given the threshold to the decision [20], where some values are decided before hand and then chosen under some criteria.

2.7 Radio Propagation Models

Propagation models are created to represent real conditions so we can evaluate solutions and predict their behaviour. When thinking about maritime communications, it usually means that we do not have many obstacles and we have a clear line of sight with favourable meteorological conditions. Also, the antenna movement when emitting or receiving a transmission, that normally its on top of mast of the ship, and more sensible to undulation movement, needs to be taken into consideration.

In previous research, some major attenuation factors where considered in radio propagation, in similar environments, such as path loss, path shadowing and multi path fading. We could consider a model with three different rays: 1) one ray direct from the sender to the receiver, 2) one ray passing through objects (like waves), 3) another ray being reflected on the surface of the sea and then to the receiver [28]. On the other hand, a simple 2 ray model, with only the direct ray and reflected ray as in Figure 2.8, has been proven to be a good approximation to the channel conditions [29, 30].

[3] https://www.mathworks.com/

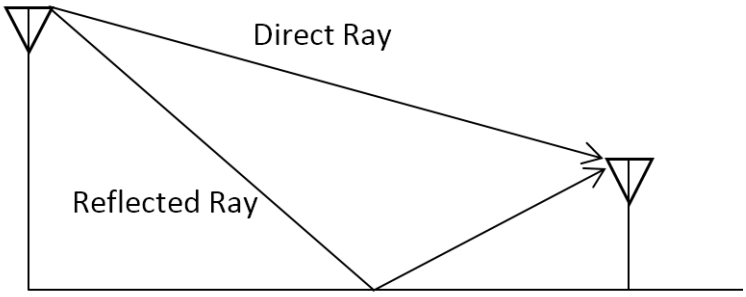


Figure 2.8: Two ray model schema.

Analyzing the model from [31] we consider only the free-space propagation loss (PL_0) and path loss caused by the reflection ray (PL_1). In this case hull penetration loss and diffraction loss caused by earth's curvature are not relevant mainly because the antenna will be placed at the top of the mast and at this distances this diffraction loss is not significant. So we have the propagations path loss as:

$$PL_{path} = PL_0 + PL_1 \tag{2.12}$$

And the free-space loss is given by:

$$PL_0 = 10log\frac{[4\pi d]^2}{G_i G_r \lambda^2} \tag{2.13}$$

where λ is the carrier wavelength in nm, d the distance between antennas in m from the land and mobile node and G_i, G_r the respective antennas gains in dB.

The reflected signal contributes to the signal loss with:

$$PL_1 = 4\sin^2\left(\frac{2\pi h_t h_r}{\lambda d}\right) \tag{2.14}$$

where h_t, h_r are the height (in m) of the antennas as the Figure 2.9 shows.

Considering that if we start to go further distances and put the antenna inside the boat, other variables will have to be taken into account as they have bigger impact in the signal losses.

2.8 Wave Movement Model

As the waves motion is one of the main causes of instability at sea, if we can model the movement, we can use that pattern to try to improve the connection. In [32] a wave motion model is described based on mathematical function, the trochoid. Trochoid mathematical functions are the ones that most resemble with the real sea undulation. The Trochoid function traces the position of one

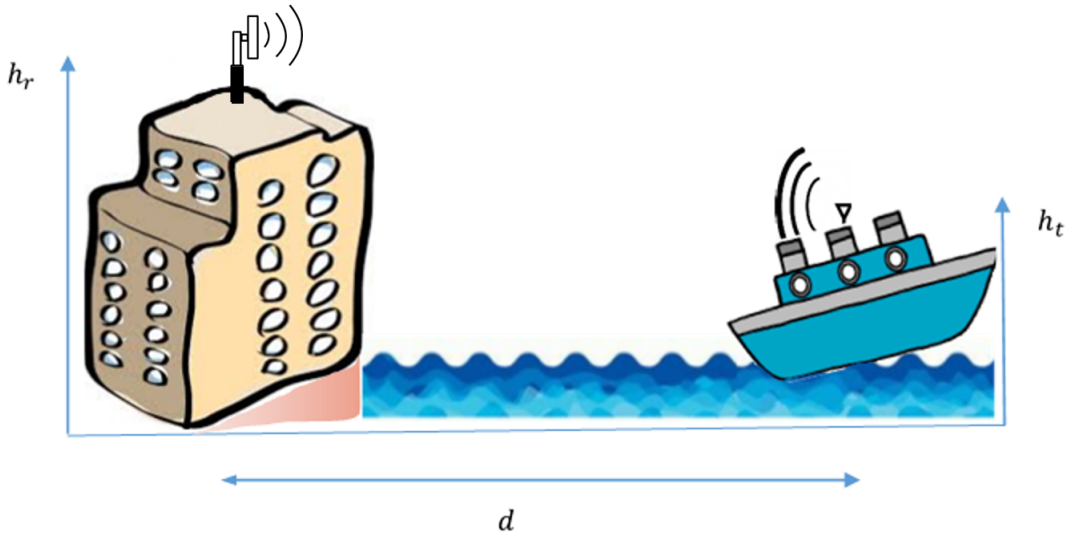


Figure 2.9: Maritime channel codelling.

point at a fixed distance of the center of the circumference, while the circumference is rolling on a transversal trajectory.

Trochoid parametric equations can be given by:

$$x = a\theta - b\sin(\theta) \tag{2.15}$$

$$y = a - b\cos(\theta) \tag{2.16}$$

As the authors of [29] state a and b can be described id terms of:

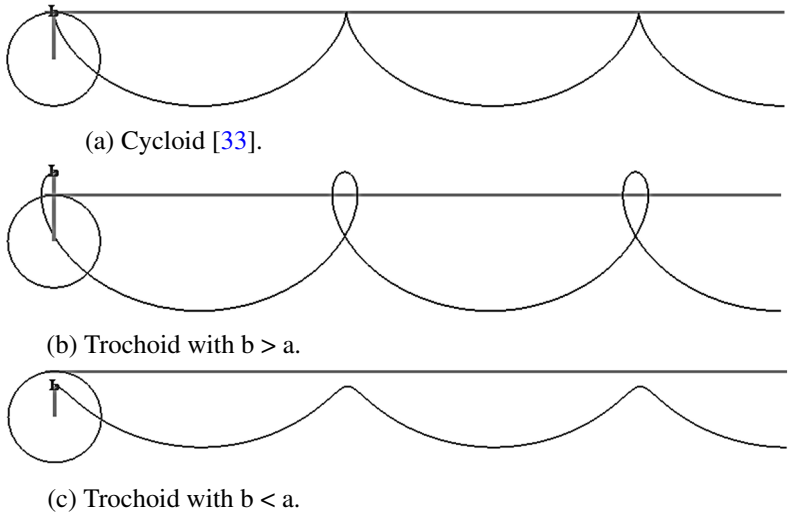
$$a = \lambda/2\pi \tag{2.17}$$

$$b = H/2 \tag{2.18}$$

with λ meaning the wavelength (in nm) and H the wave height, in m. They also simplified θ so it could be:

$$\theta = 2\pi(\frac{x}{\lambda} - \frac{t}{T}) \tag{2.19}$$

where x is the distance from the ship to land base, T is the wave period and t represents the time instant.



(a) Cycloid [33].

(b) Trochoid with b > a.

(c) Trochoid with b < a.

Figure 2.10: Trochoid examples

Representing trochoid equations we can have several scenarios. A special case is 2.10a where a has the same value as b. If we have b > a the movement results is something like Figure 2.10b. But if instead b < a the trochoid line is more like Figure 2.10c.

2.9 Summary

Presently there are some maritime communications solutions, but they are still very expensive or complex. Wi-Fi is one of the most cost-effective alternative to provide broadband communication near shore. In this chapter were also presented some work related to the variation of the MTU and some of the most widespread rate algorithms, showing that there is some improvement opportunities for an unsteady channel. We approached the main causes for losses in a wireless communication. A study was also made regarding the propagation of radio waves and the characterized distinctive maritime conditions like the movement of the sea surface.

Chapter 3

Proposed Solution

In this chapter, we present the proposed adaptive algorithm, which dynamically adjusts the data rate and MTU, based on RSSI, to maximize throughput. A crucial component of the algorithm is how the data rate and MTU values are selected. This chapter presents how that choice is made and the ARMA mechanisms.

3.1 Ideal MTU and Rate Values

From the previous study of MAC and PHY layers from IEEE 802.11 in Section 2.2, we started with a theoretical approach on the throughput (Equation 3.1), based on the IEEE 802.11 Distributed Coordination Function (DCF) for the accordingly IEEE 802.11 protocol. We considered the useful data with MTU size in bytes (we multiply by 8 to obtain the resulting value in Mbit). The total time needed to transmit one packet is the sum of the time to transmit all overhead associated, the time to transmit the payload (of MTU size at the physical data rate) and the time of the acknowledgement. So the maximum theoretical throughput can be defined has the payload size over the total time it takes for it to be transmitted (Equation 3.1).

$$Maximum\ theoretical\ throughput\ = \frac{MTU\,(in\ Bytes)\ x\ 8}{Data\ Time\ +\ Overhead\ Time\ +\ ACK\ Time} \tag{3.1}$$

For the different data rates the time it will take to transfer the same amount of information differs: it takes less time to higher data rates to transmit the same information. Considering the same amount of overhead, the bigger the MTU size the better. We want to maximize the MTU reducing the overall overhead, increasing the throughput. We began to consider an ideal channel to understand how the maximum theoretical throughput could vary with the increasing MTU and different data rate. Consequently, for each data rate we incremented the MTU and calculated the respective theoretical throughput and constructed the graphic in Figure 3.1, with a continuous set of MTU values.

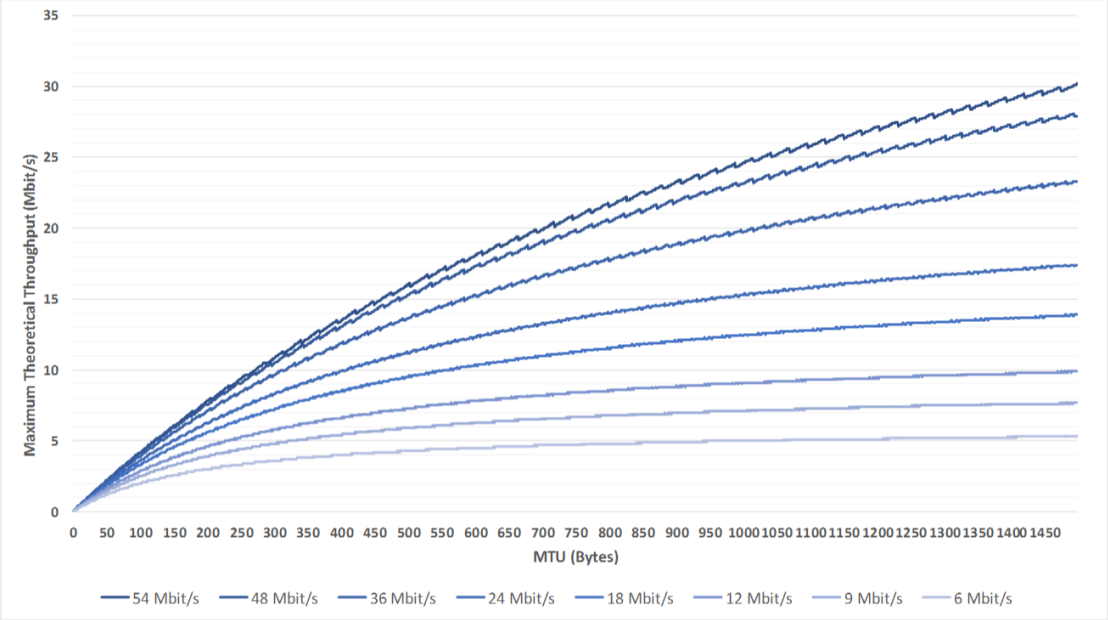


Figure 3.1: Maximum Theorethical Througput Varation for Diferent Values of Data Rate and MTU.

In order to maximize the throughput, we traced an imaginary curve with gradient 0 for each throughput value. The interception with each data rate curve gave us the a match (MTU,data rate) for that throughput. In order to optimize the selection we chose the lowest data rate and highest MTU in case there was more than on intersection point. This way the transition point from one data rate to the next would be the minimum MTU in the next data rate that could achieve higher throughput.

From these results, we realized that changing only small amounts the MTU was not efficient. For our case we considered discreet values of MTU from 500 to 1500 bytes. For each data rate we obtained a new curve for the throughput variation in order of new MTU values, as seen in Figure 3.2. If we consider different MTU values we would obtained different curves.

For example, to achieve higher throughput after 36 Mbit/s and MTU of 1500 we had to proceed to the net higher data rate, 48 Mbit/s. But we can only achieve higher throughput for 48 Mbit/s with a MTU bigger than 1200, from our pool of choices. After this we increase the MTU until we ascent to the next data rate. In Figure 3.2, at black, we have the optimal matches (MTU, data rate) variation to maximize throughput, for our selection of MTU values.

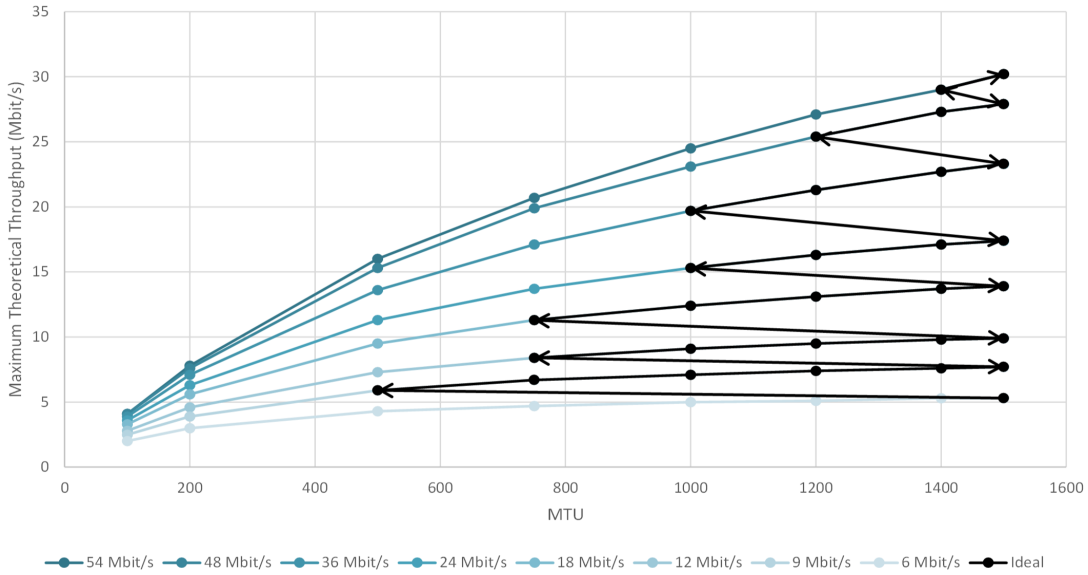


Figure 3.2: Maximum theorethical througput varation for diferent values of data rate and MTU with discrete values of MTU.

The higher the rate the more information it can be sent in the same time interval. For each rate the bigger the MTU, the more efficient the information transfer will be, because we have the same amount of overhead for more effective data. Those ideal thresholds are the values where the next rate with some smaller MTU value has a higher throughput. This means more useful data in the same time interval. As expected, we can only obtain higher throughput with higher data rates and bigger MTU.

As presented in Chapter 2 a physical channel introduces more complexity, adding an error probability. In reality, we have errors. So, the throughput value, in 3.2, will have to take into account the Packet Loss Probability (P_L).

$$Calculated\ Throughput\ = Maximum\ Theoretical\ Throughput\ \cdot\ (1-P_L) \tag{3.2}$$

Considering the throughput as the amount of frames that successfully arrived at the destination with the chosen data rate. For this solution we consider a point-to-point link, disregarding frame collision (see Section 2.7) so $P_L = P_e$. Considering no errors occurs in the header transmission, we can define (P_L) (see Section 2.7) as:

$$P_L\ =\ P_e\ =\ 1\ -\ (1\ -\ BER)^L \tag{3.3}$$

BER is dependent on the modulation, as explained in Section 2.7. Nevertheless, BER is a function of E_b/N_0. As E_b/N_0 can be represented as:

$$\frac{E_b}{N_0}\ =\ SNR \cdot \frac{Bandwidth}{Data\ Rate} \tag{3.4}$$

RSSI and noise values give us information about the state of the channel and we can use them to calculate SNR.

$$SNR = \frac{RSSI}{Noise} \tag{3.5}$$

Using RSSI and noise as input, we can calculate the MTU and data rate that maximizes throughput. With the RSSI and noise values we can calculate the SNR with Equation 3.5. With SNR we retrieve BER with Equation 3.4 and Equations from Section 2.6 obtaining P_L from there with Equation 3.3. Keep in mind the pair (MTU, rate) is obtained comparing the throughput value from all ideal pairs for the specific RSSI and noise values, selecting the highest one.

In the Table 3.1 is shown an example of the best MTU and data rates is shown (in IEEE 802.11a protocol) for different RSSI values, fixing noise at -95 dBm.

Table 3.1: MTU and data rate chosen for several RSSI values with noise fixed at -95 dBm

RSSI (dBm)	Data Rate (Mbit/s)	MTU	MODULATION
-67	54	1500	64-QAM
-72	54	1400	64-QAM
-73	48	1200	64-QAM
-79	36	1000	16-QAM
-80	24	1000	16-QAM
-86	18	750	QPSK
-88	12	750	QPSK
-90	9	500	BPSK
-94	6	1500	BPSK

3.2 Adaptive Rate and MTU Algorithm (ARMA)

Holding the main guidelines for the algorithm we had to adapt it to a maritime environment. The fact that we read the RSSI value from the OS has its limitations since it can only be read once every second. As in this particular environment the SNR is constantly fluctuating, we had the need to address this problem. We decided to use the proposed RSSI prediction algorithm presented in Section 2.4 that manages to find a solution for both problems: 1) it predicts the RSSI value between the time intervals, 2) it is also adapted for a maritime environment using a sinusoidal model so emulate the channel state variation.

For the implementation of the design presented in Section 3.1 we developed Algorithm 1. It is a never ending cycle continuously reading the most recent RSSI value from the OS. In the time between readings we use the prediction algorithm that estimates the RSSI value. We use this information as input to the change_mtu_rate function posed in Figure 3.3. This function computes the RSSI and noise values for each ideal pair (MTU, rate), calculating the intermediate data, comparing the final throughput value and opting for the highest if the P_L is lower than a

defined threshold. We decided to introduce this threshold because sometimes the (MTU, rate) we chose had a much higher P_L than the next option but the calculated throughput was only slightly higher. By default the threshold value is 6 %, we came to this value through an empirical analysis and applied it whenever it was possible.

In each call the function applies the best MTU and rate combinations if they are different from the ones in use, increasing the efficiency.

Algorithm 1 Adaptive Rate and MTU Algorithm (ARMA)

```
procedure START
    while TRUE do                          ▷ At the start need 4 seconds to seed the vector
        if Time > Prediction Time then     ▷ Read RSSI from OS
            Read MTU
            Read RSSI
            Read Noise
            Update RSSI Vector
            Update positions wave
            change_rate_mtu(RSSI,Noise)    ▷ Call Change Rate and MTU
                                           ▷ with real RSSI value
        else
            Read Noise
            Predict RSSI value
            change_rate_mtu(RSSI,Noise)    ▷ Call Change Rate and MTU
                                           ▷ with predicted RSSI value
        Update Prediction Time
```

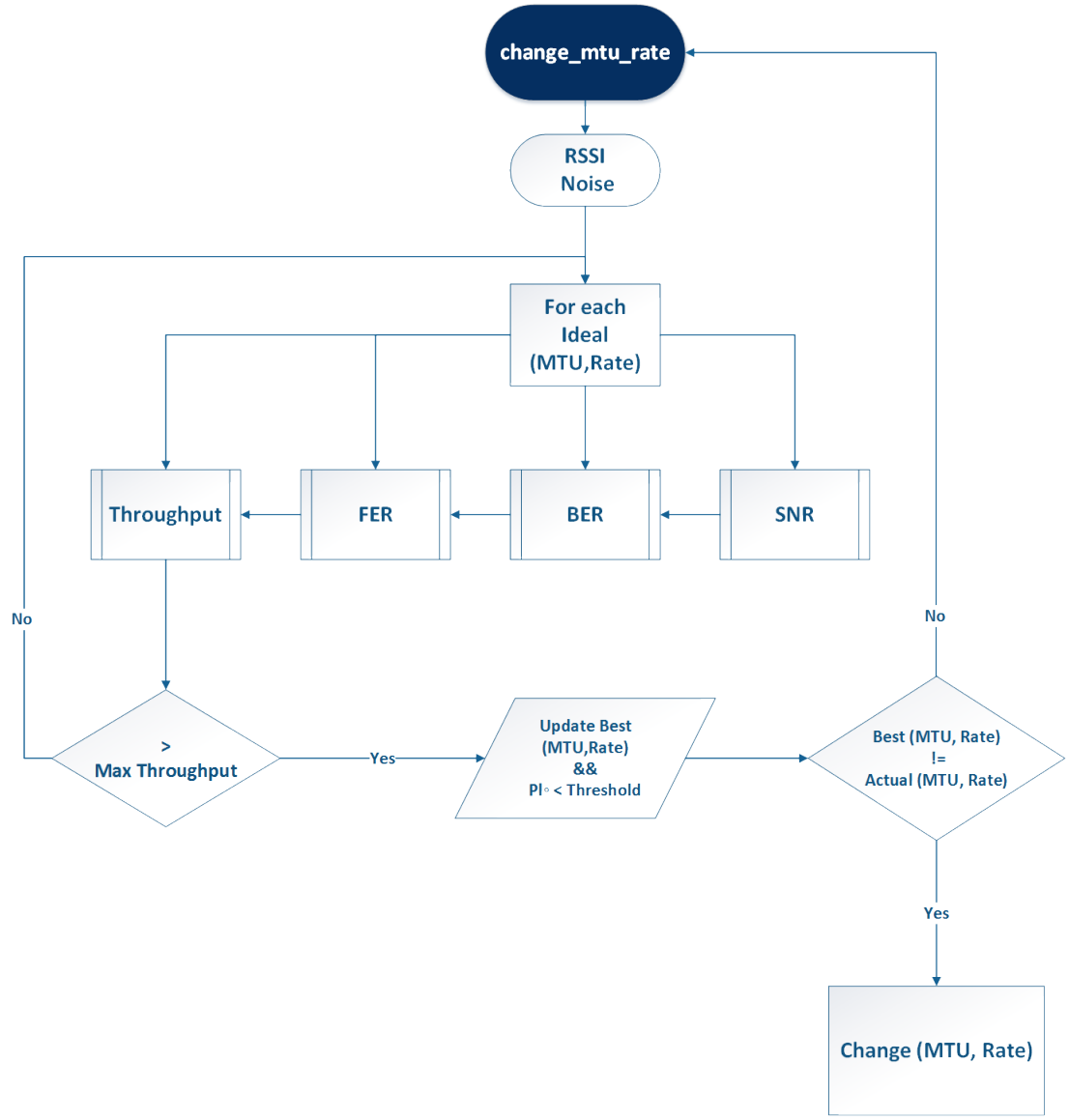


Figure 3.3: change_mtu_rate Function Diagram.

Chapter 4

Implementation and Prototype

Having presented the concept of ARMA algorithm in Chapter 3, in this chapter, we describe how the implementation was carried out. The essential hardware and software for the the prototype is displayed. We also approach the parameters to be evaluated in the tests performed.

4.1 Proposed Solution Implementation

The conception of the algorithm presented in Chapter 3 was done in synchrony with its implementation in a step by step cyclic process. The results obtained from the implementation were used as feedback to gradually adjust the final algorithm. In this section, we address the conditions in which the proposed solution was implemented. We develop a program, written in C language to run on the OS, OpenWRT.

In order to retrieve the MTU information we use a well known system tool, *ifconfig* invoking the command (4.1), and parsing the desired information.

$$ifconfig\ wlan0 \mid grep\ MTU \tag{4.1}$$

To fetch the remnant noise and RSSI values we had to resort to a more specific tool, *iwinfo*, calling the command (4.2) and isolate the pertinent information.

$$iwinfo\ wlan0\ info \tag{4.2}$$

We encountered a critical problem when managing the current rate in use. The value of the rate read from the wireless card was not always accurate. Tests were made to find a correlation between the value read from the OS and a rate value forced by us. As a conclusive relation was not achieved, we circumvent this issue by assigning a variable with the last rate value forced by us. To make sure the rate was in fact changing, we performed tests generating traffic with Iperf and simultaneously forcing different rate values with 4.4. Then we observed the effective throughput change. And as can be seen in Figure 4.1 the throughput did change, the rate value read from the OS did not, at least as expected.

```
[  3] local 192.168.2.71 port 5002 connected with 192.168.2.70 port 60249
^Croot@Barco:~#
root@Barco:~#
root@Barco:~#  iperf -s -u -i 1 -p 5002
------------------------------------------------------------
Server listening on UDP port 5002
Receiving 1470 byte datagrams
UDP buffer size:  160 KByte (default)
------------------------------------------------------------
[  3] local 192.168.2.71 port 5002 connected with 192.168.2.70 port 33475
[ ID] Interval       Transfer     Bandwidth        Jitter   Lost/Total Data
grams
[  3]  0.0- 1.0 sec   636 KBytes  5.21 Mbits/sec   3.696 ms    0/  443 (0%)
[  3]  1.0- 2.0 sec   636 KBytes  5.21 Mbits/sec   6.488 ms    0/  443 (0%)
[  3]  0.0- 2.2 sec  1.34 MBytes  5.21 Mbits/sec   7.303 ms    1/  960 (0.1
%)
[  3] local 192.168.2.71 port 5002 connected with 192.168.2.70 port 44325
[  3]  0.0- 1.0 sec   913 KBytes  7.48 Mbits/sec   3.654 ms    0/  636 (0%)
[  3]  1.0- 2.0 sec   913 KBytes  7.48 Mbits/sec   1.853 ms    0/  636 (0%)
[  3]  0.0- 2.1 sec  1.85 MBytes  7.48 Mbits/sec   5.118 ms    1/ 1319 (0.0
76%)
[  3] local 192.168.2.71 port 5002 connected with 192.168.2.70 port 59275
[  3]  0.0- 1.0 sec  1.15 MBytes  9.63 Mbits/sec   1.757 ms    0/  819 (0%)
[  3]  1.0- 2.0 sec  1.15 MBytes  9.63 Mbits/sec   1.909 ms    0/  819 (0%)
[  3]  0.0- 2.1 sec  2.38 MBytes  9.63 Mbits/sec   3.926 ms    1/ 1700 (0.0
59%)
[  3] local 192.168.2.71 port 5002 connected with 192.168.2.70 port 44354
[  3]  0.0- 1.0 sec  1.61 MBytes  13.5 Mbits/sec   1.903 ms    0/ 1147 (0%)
[  3]  1.0- 2.0 sec  1.61 MBytes  13.5 Mbits/sec   1.231 ms    0/ 1149 (0%)
[  3]  0.0- 2.0 sec  3.30 MBytes  13.5 Mbits/sec   2.822 ms    1/ 2352 (0.0
43%)
[  3] local 192.168.2.71 port 5002 connected with 192.168.2.70 port 42856
[  3]  0.0- 1.0 sec  2.00 MBytes  16.7 Mbits/sec   0.883 ms    0/ 1424 (0%)
[  3]  1.0- 2.0 sec  2.00 MBytes  16.8 Mbits/sec   0.867 ms    0/ 1427 (0%)
[  3]  0.0- 2.0 sec  4.07 MBytes  16.8 Mbits/sec   2.135 ms    1/ 2907 (0.0
```

Figure 4.1: Effective throughput change as we alter the physical rate from 6 to 9, 12, 18, 24 Mbit/s

The rate, noise and RSSI are used as input to the *change_mtu_rate* function (Figure 3.3). While in this function, accordingly to the algorithm, for each of the ideal rate and MTU pair, we calculate the SNR and BER and respective throughput. If the throughput is higher that the current maximum and the P_L does not exceed the threshold, this pair of MTU and rate will be assigned as the new best match. If the new best match is different from the current rate and MTU we enforce the respective MTU using the command (4.3)

$$\textit{ifconfig wlan0 mtu } x \tag{4.3}$$

where x represents the selected MTU value.

We also update the rate value forcing the chosen value with (4.4)

$$\textit{iw dev wlan0 set bitrates legacy-5 } y \tag{4.4}$$

where y represents the selected rate to be used, varying from 6 Mbit/s to 12, 9, 18, 24, 36, 48 to 54 Mbit/s .

Our code includes specific functions aimed at increasing the reliability by checking for failing readings or in case the connectivity is lost it, waits till it is restored, reporting the error.

4.2 Hardware

In this section we talk about the hardware components used for testing the algorithm.

- **System Board** – This system board is composed by an Alix3d3 (Figure 4.2). It has a 500MHz AMD Geode LX800, with x86 architecture and 256MB DDR DRAM. It also has two USB ports and is powered by a Power Over Ethernet (POE) port. POE allows to power and transmit data at the same through an Ethernet cable;

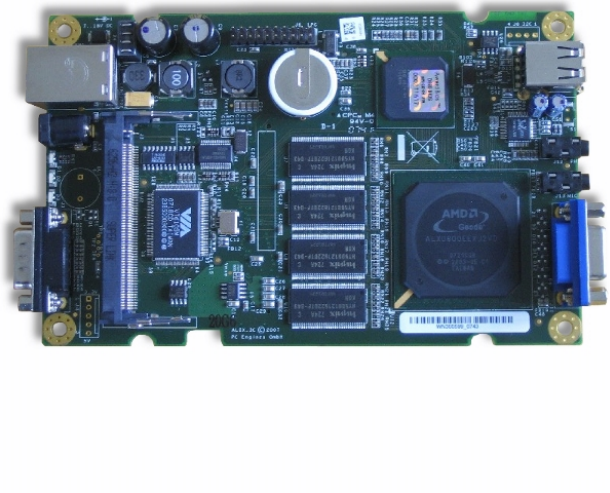


Figure 4.2: Alix3d3 Board.

- **Wireless Card** – The wireless card attached to the board is a Mikrotik R52n-M 802.11a/b/g/n MiniPCI card (Figure 4.3). This card can operate in both 2GHz and 5GHz bands. It supports output power up to 23dBm. It also supports different modulations including OFDM: BPSK, QPSK, 16 QAM, 64QAM and DSSS: DBPSK, DQPSK. It has a high performance, being able to reach up to 300Mbps physical data rate.

Figure 4.3: Mikrotik R52n-M IEEE 802.11 wireless card.

- **Antennas** – For laboratory tests were used a RF Elements StationBox Mikro (Figure 4.4) with 5GHz 15dBi dual polarization directional antenna, with a radiation diagram [1] as in Figure 4.5,
  Two different models will be used on MARBED testbed. One in land will be a Ubiquiti AM-5G16-120 with 16dBi gain. The material on the boat have un Ubiquiti AMO-5G10 with 10dBi gain;

Figure 4.4: RF Elements StationBox Mikro directional antenna.

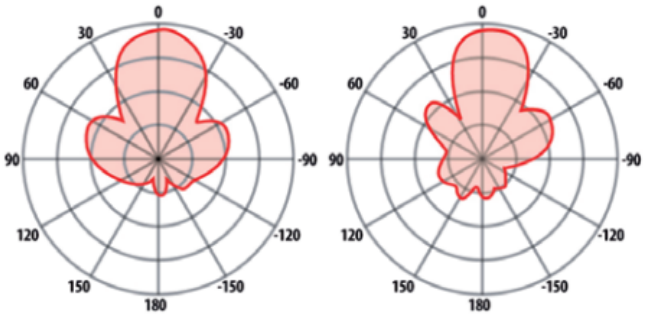


Figure 4.5: 5 GHz, 15 dBi directional antenna radiation diagram.

- **GPS Module** – The system will also have a GPS for tracking. The module that will be used the USGlobalsat BU-353 plug and play device with no need for external power supply. This device data was handled by specific software presented in the following section;

[1] Antenna datasheet: http://varia-store.com/media/products/0155894001348729194.pdf

4.3 Software

To run the tests we used related software including specific Operative System and wireless communication tools.

- **OpenWRT** – Is a Linux distribution specialized for embedded systems. It provides a good level of customization and management for key components like some of wireless card variables. It is also adequate for a low resources system like ours and compatible with several useful tools that will help us retrieve the variables taken into consideration;
- **Eclipse** – Eclipse is an Integrated Development Environment (IDE) where we develop our code in C language. Also, once our Alix board doesn't fulfil the minimum requirements to support a full C compiler we also had to cross compile our code on a x86 computer;
- **Iperf** – An application that provides a bundle of tools to generate traffic. Is can be used to measure bandwidth performance and TCP and UDP parameters. It also can provide information about delay and RTT, jitter and datagram loss;
- **Gpsd** – Gpsd is a commonly used service daemon for Linux systems. It manages data collected from GPS sensors and can be used with a large variety of devices;
- **Other Wireless Tools** – Other wireless core utilities provided by OpenWrt software. For example, *iwconfig*, that allows us to see several data about the wireless card related data;
- **NTP/ntpdate** – These are deamons that allow us to create a time server and update the clock time through a server so multiple nodes can be synchronized.

4.4 Evaluated Metrics

For the correct evaluation, testing, comparison and data collect we need to restrict to the most important variables. Later we will compare this metrics for the presented algorithm and the most generally used.

- **Throughput** – It is the transmission rate of data per time period. This is, how much data we send per time unit, normally in Mbit/s. We will mainly focus on this metric once our solution was thought to maximize this measurement;
- **RSSI** – signifies the power level of the signal received by the antenna, also measured in dBm. This gives us an idea od the state of the channel. The higher the RSSI value, the better the signal.

4.5 Network Topology

The fundamental test scenario, showed in Figure 4.6 has two different nodes: 1) representing the Server, usually the land station and 2) the Client node, a dynamic node representing the boat, for example.

Figure 4.6: Basic network topology.

This is the basic network topology in the test scenarios. Different test may add a different layer of complexity to this design, accordingly to the test performed. We may or may not be able to have a physical connection to each node but we are able to access each node through the other if a link between both nodes exists. This connection can be done through a Secure Shell (SSH), which even allows us to control both nodes remotely.

4.6 Test Runs Scripts

Given that the test case is not suitable for a manual operated system, we created Shell Scripts in order to obtain a fully automated system, what is also a requirement for the testbed. To make a more robust and flexible system we created different modules of scripts (Figure 4.7) for easy adaptability. With this, we can remove or add new functionalities as needed and even test singular modules individually. For the scripts to work accordingly, both nodes had to be synchronized. Having the corresponding scripts, in the different nodes, starting the same time concede is necessary. In order to do this, we had to setup a Network Time Protocol (NTP) between both nodes. The land node would act as the NTP server by which the mobile node would synchronize, with *ntpdate*, as soon as the connection between them was working. The following flowcharts (Figure 4.7) represents the chronological tasks being processed by each node in parallel. To facilitate the process of retrieving and separating data each test occurs in 30 second interval.

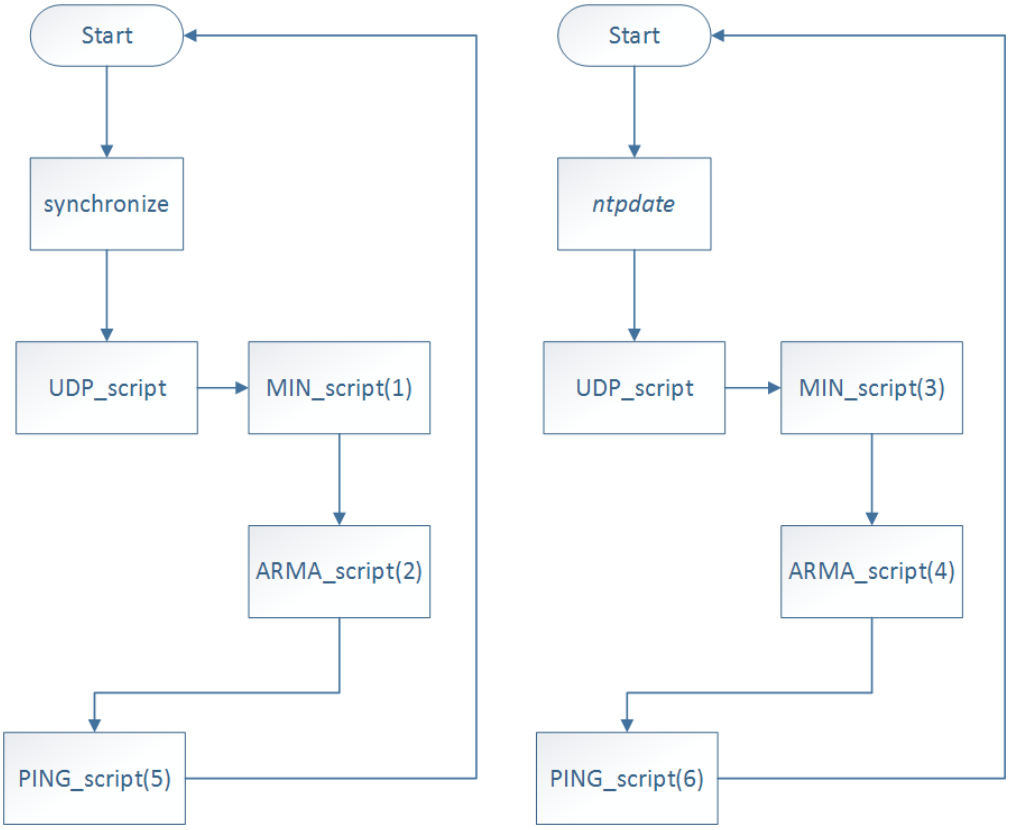


Figure 4.7: Server (left) and Client (right) scripts schema.

- **(1) and (2)** - In this script, on the server side, we start the Iperf UDP and read the RSSI values continuously.
- **(3) and (4)** - On the client side, we first start running the Minstrel (3) and ARMA (4) algorithms, respectively. After that, Iperf will generate traffic for 25 seconds. We gave some leverage to prevent eventual delays. At the same time we register the RSSI values read and the GPS coordinates.
- **(5) and (6)** - After the traffic test we initiate a ping test both from Server-Client and Client-Server recording the RSSI values first for the Minstrel and second for ARMA.

Each individual process generates a unique file that will be saved and organized according to the data that it contains. The data will be separated in folders *(type_of_data)_(algorithm)_(date)*. The files that contain data will be named after (date)_(file_extension), where we have a different file extension for each data.

For example, we would have a folder PING_*date* that contains the files *date*.ping_min_s and *date*.rssi_s on the server side. Other example is, on the client side, a folder UDP_ARMA_*date* that contains the files *date*.iperf_arma_udp_c, *date*.gps_arma_udp_c and *date*.rssi_arma_udp_c .

4.7 Data Retrieving

In order to retrieve useful data from the logs created, we used other Shell Scripts in collaboration with AWK[2]. AWK is a program that can select specific records on a file and manage them. This program was used to parse and clean the logs excluding non significant information. This was made in a posterior time after the logs were collected to save processing time, dedicating the full processing time to retrieve data. On the other hand, postpone logs parsing imply increase space taken by all the logs files unprocessed.

In Figure 4.8 is shown the flowchart for data retrieving. Starting with the time interval that will be used in folders and files names, this time will be increased each step until it reaches the MAX value selected. Each iteration checks the existence of a folder and files corresponding to the time variable If it exists, this data will be grouped if a file containing all the information for the same type of data. When the MAX value is reached the final file containing all grouped data for each file is cleaned remaining only useful information.

The final step is to aggregate the corresponding data for analysis, meaning to join the RSSI values and the corresponding throughput. The final file will contain the comparison between equivalent type of data for each test done.

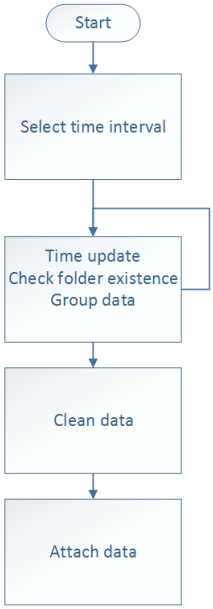


Figure 4.8: Data parsing and retreiving scripts schema.

The final file containing the aggregated data will be processed using Microsoft Office Excel[3]. With this program we created the tables and graphics that summarize the collected information creating pertinent statistics related to that data.

[2]https://www.gnu.org/software/gawk/manual/gawk.html
[3]https://products.office.com/en-us/excel

Chapter 5

Experimental Evaluation

The experimental setup and tests performed to evaluate the ARMA algorithm are presented in this chapter. We exhibit with more detail the different tests carried out and explain the major configurations and different environments where they were performed. This chapter also includes the obtained results and its analysis.

5.1 Experimental Setup

For each test, we always had at least two different systems nodes composed by a bundle of an Alix3d3 board associated with a MikroTik R52n-M wireless card and one respective antenna as described in Section 4.2.

Initially, we used IEEE 802.11n. However, when we tried to force the physical data rate the values did not change, maybe due to driver problems. This made IEEE 802.11n not reliable for this case, so we decided to adopt IEEE 802.11a instead. The IEEE 802.11a protocol had a more robust implementation in the OS used, which allow us to manipulate relevant variables with more ease. We chose the 5.8 GHz band because it is a worldwide license-free band and showed good results in previous maritime communications work [3, 4].

In all the tests, we take into account the premise that the channel is symmetric. This implies that we consider that both nodes see the same channel state and quality and thus be equivalent for both nodes.

As our goal was to maximize the raw throughput we could reach in a point-to-point connection. We opted to analyse only the traffic generated using UDP. UDP minimizes the overhead compared, for example, with TCP as it has no congestion control mechanisms. It also works well in unidirectional communication, like our case, where there is only one node sending information at the time.

5.2 Laboratory Evaluation

The introductory tests were conducted in a laboratory controlled environment, having each node in line-of-sight at a distance of approximately 10 *m*. In this setup, the server node had an omni-directional antenna with 5 *dBm* gain. The client node had a directional antenna (see Subsection 4.2). The directional radiation of this antenna lead us, to a certain degree, control the amount of signal strength of the connection. Huddle together with a semi-circle movement (illustrated by Figure 5.1) we try to emulate similar conditions to those faced at sea, assuming the RSSI variation follows a sinusoidal pattern.

The information gathered in this tests lead us to hone the succeeding real environment test cases.

Figure 5.1: Stop motion representation for the antenna movement.

5.2.1 Laboratory Tests

In this laboratory tests we took a similar approach to the topology presented in Section 4.5. Adding an additional interface to each node in a different VLAN granting us remote access. The topology can be represented by Figure 5.2

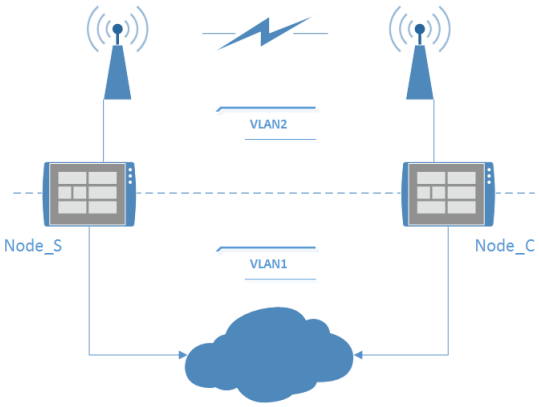


Figure 5.2: Laboratory tests network topology.

5.2.2 Evaluation Results

For the initial laboratory tests we only compile data regarding the throughput. We did this to start to understand what would be the impact of our algorithm. We gathered the information collected from the tests and grouped the mean throughput value for each corresponding RSSI value. So being, we composed for each test one table for ARMA and used the Minstrel algorithm as the reference for the comparison. We concentrated the information analysing the mean throughput obtained for each RSSI value. In result, we obtained the information presented in Table 5.1 for ARMA and Table 5.2 for Minstrel.

The first results were positive and indicated a mean gain in throughput using the ARMA algorithm comparing with Minstrel. As we can see in Table 5.1 for a signal with the mean RSSI of -86 dBm, for example, we have a mean throughput of 6,17 Mbit/s for ARMA while for Minstrel is only 4,38 Mbit/s, which means an increase of more than 40%.

Table 5.1: Collected data synopsis for ARMA.

RSSI (dBm)	**#N**	**Mean (Mbit/s)**	**StDev**
-73	14	10,33	3,16
-74	15	9,95	2,74
-75	21	11,46	2,15
-76	46	11,91	2,35
-77	47	10,52	2,91
-78	44	9,75	2,51
-79	26	8,24	3,06
-80	29	8,35	3,09
-81	29	8,81	1,74
-82	159	6,32	1,39
-83	3727	6,21	1,31
-84	6616	6,25	1,27
-85	1987	6,23	1,34
-86	**522**	**6,17**	**1,23**
-87	500	6,00	1,11
-88	447	5,95	1,19
-89	203	5,94	1,01
-90	63	5,94	0,71
-91	4	6,06	0,34

Table 5.2: Collected data synopsis for Minstrel.

RSSI (dBm)	**#N**	**Mean (Mbit/s)**	**StDev**
-73	10	10,03	0,99
-74	19	12,88	2,13
-75	33	12,68	2,62
-76	44	11,85	3,03
-77	29	10,64	3,67
-78	40	9,90	2,85
-79	37	9,06	2,04
-80	30	7,78	2,52
-81	30	7,80	2,43
-82	262	6,23	1,48
-83	4925	6,06	1,34
-84	8757	6,02	1,27
-85	2366	5,62	1,40
-86	**554**	**4,38**	**1,88**
-87	627	4,93	2,38
-88	586	4,97	2,45
-89	261	5,10	2,41
-90	63	4,68	2,54
-91	9	4,91	2,00

The graphic in Figure 5.3 shows the mean throughput in order to the RSSI, with 95% confidence values. As we can see, for good signal strength the Minstrel algorithm has a better behaviour than ARMA. On the other hand, as the RSSI value decreases, ARMA maintain a steady higher mean throughput.

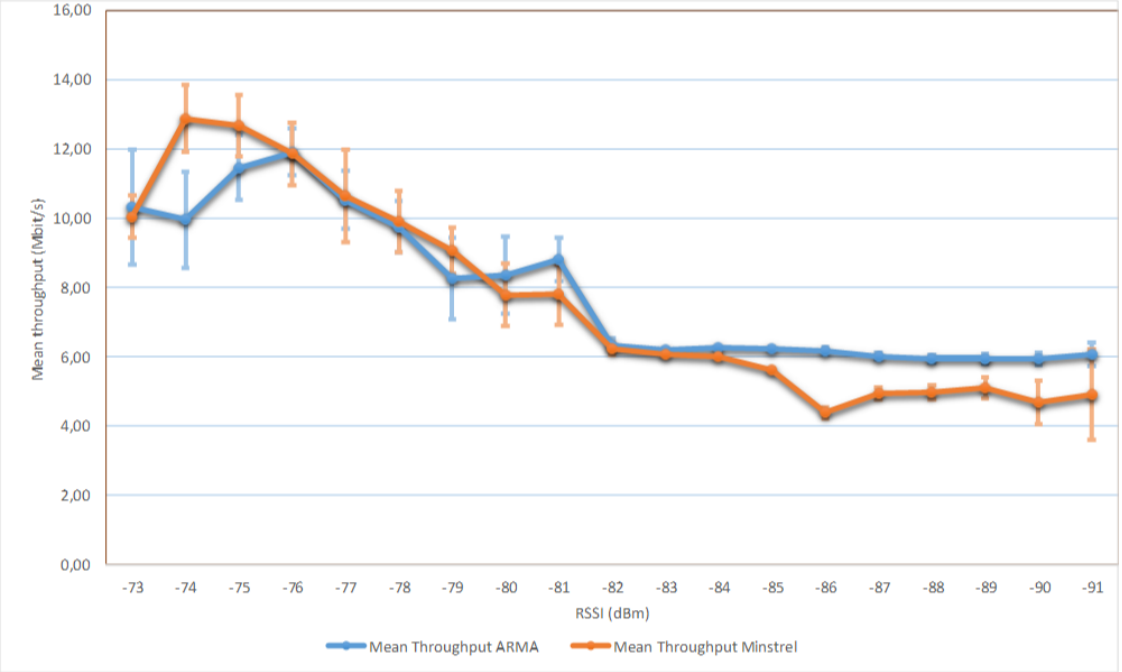


Figure 5.3: Graphic comparing the mean throughput variation with RSSI for ARMA and Minstrel at the laboratory.

For a more generic comparison considering the Mean Gain (Equation 5.1) as:

$$Mean\ Gain_{\ RSSI}\ (\%) = \left(\frac{Mean_{ARMA\_RSSI}}{Mean_{Minstrel\_RSSI}} - 1 \right) \cdot 100 \tag{5.1}$$

We can obtain the mean gain for each RSSI received (accordingly to Equation 5.1). It is summarized in Table 5.3 the results obtained. We obtained up to a 40% mean gain improvement using ARMA instead of Minstrel, for the maritime conditions simulated.

The under-performing results for ARMA when having a signal with higher RSSI can be explain by considering that ARMA was designed for a very dynamic environment. Usually when having a good signal, the RSSI variations are not so relevant because the link is healthy. In this case adapting the rate and MTU can be counter-productive. As our algorithm is expecting the link state continuously to be changing, it will react to the slightly variation when in reality it would not be necessary.

Table 5.3: Mean throughput gain comparing ARMA to Minstrel in laboratory tests

RSSI (dBm)	Mean Gain %
-73	**2,93**
-74	-22,74
-75	-9,59
-76	**0,48**
-77	-1,11
-78	-1,46
-79	-8,98
-80	**7,41**
-81	**12,94**
-82	**1,35**
-83	**2,45**
-84	**3,89**
-85	**10,90**
-86	**40,90**
-87	**21,73**
-88	**19,74**
-89	**16,62**
-90	**26,84**
-91	**23,39**

5.3 Robotics Exercise 2015 (REX15)

This event was fostered by the Portuguese Navy Research Center in partnership with the Portuguese Naval Academy and was held near the Portuguese Naval Base in Alfeite. We were able to test for the first time the proposed solution in a real maritime environment, although we still had some control over the tests. Here we improvised main workbench with the land node, as the server. We also had a small craft available to carry the mobile node, as the client, in the Tejo estuary (Figure 5.4).

Figure 5.4: Close up with both land node and the small craft carrying the second node.

The tests were performed with the node in the boat having the movement imposed by the water currents. This lead to the link being broken a few times and the connection to be re-associated.

In the functional tests we noticed a discrepancy of the channel state between the nodes that was overtaken by adjusting the Transmission Power on one node. Later it was solved by changing some hardware components.

For this test we also add an additional component in order to prove the applicability of the algorithm: we attached a camera to the client node and streamed a small video to prove the usability in a real application. The frame in Figure 5.5 was obtained through the video transmission of Lisbon shore, that proved fluid, only noticing some loss as the signal worsens.

Figure 5.5: Video stream frame.

5.3.1 REX15 Evaluation Setup

In the REX15 setup we could not use the GPS module as the Alix board only has 2 USB ports and one was occupied by the booting pen and the other was where the video camera was connected. In this test case (Figure 5.6) we separated the video camera in a different VLAN for easy traffic routing. This way our system in VLAN2 just acts as a black box for the traffic generated in VLAN3 by the video stream and sent to VLAN1 to a laptop client.

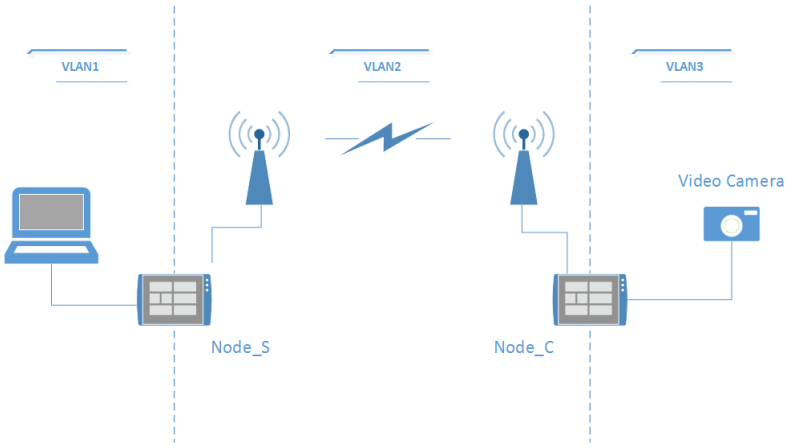


Figure 5.6: REX15 tests network topology.

5.3.2 RSSI Prediction Algorithm Validation

During this test we also recorded a graph with the RSSI variation over time recorded through the graphic Web User Interface interface of our system, LuCi[1]. In Figure 5.7 we can easily identify a form that resembles a sinusoidal wave. Thus we could validate the proposed sinusoidal model in [3] and also adopted for our proposed solution.

Figure 5.7: RSSI variation over time in a REX15 test.

5.3.3 Evaluation Results

We proceeded with a similar analysis from de data collected from REX15, related to throughput. In Tables 5.4 and 5.5 we have a summary of the data from ARMA and Minstrel, respectively. We noticed that we have a considerable less number of samples, which could affect the standard deviation. We still have higher mean throughput for higher RSSI values but comparing to the equivalent tables from the laboratory tests we observe the means throughput values to drop to all RSSI values.

[1] http://wiki.openwrt.org/doc/techref/luci

Table 5.4: Collected data synopsis for ARMA at REX15.

RSSI (dBm)	#N	Mean (Mbit/s)	StDev
-73	9	5,05	1,21
-74	11	4,29	1,43
-75	13	3,59	2,06
-76	6	3,98	1,21
-77	10	4,99	2,72
-78	19	4,47	2,74
-79	15	4,35	2,48
-80	24	4,05	2,59
-81	34	3,42	2,03
-82	24	2,13	2,00
-83	11	2,77	1,55
-84	19	1,70	1,82
-85	11	0,31	0,55
-86	6	0,13	0,14
-87	3	0,20	0,00
-88	1	0,29	0,00

Table 5.5: Collected data synopsis for Minstrel at REX15.

RSSI (dBm)	#N	Mean (Mbit/s)	StDev
-73	6	4,16	0,12
-74	10	4,54	0,64
-75	17	5,03	0,89
-76	7	5,70	0,81
-77	11	5,31	0,91
-78	7	5,68	1,51
-79	14	3,14	2,24
-80	30	3,00	1,99
-81	38	2,16	1,51
-82	15	1,85	0,68
-83	15	1,51	0,53
-84	14	1,50	0,75
-85	16	2,05	0,81
-86	7	2,13	0,64
-87	7	2,00	0,39
-88	2	1,50	0,10

As we can examine from Figure 5.5 with the throughput variation results obtained with a 95% confidence. We still have a superior mean throughput at some values of RSSI. As the RSSI values lower, we see the ARMA performance overtake the Minstrels, in the interval of RSSI between -79 and -84 dBm. But on the other side, at very lower RSSI we obtained a worse trend than Minstrel. This difference with the laboratory results may have to do with some difference in the environmental conditions when the tests were performed. Having a low number of samples for this RSSI values do not allow us to make a accurate comparison.

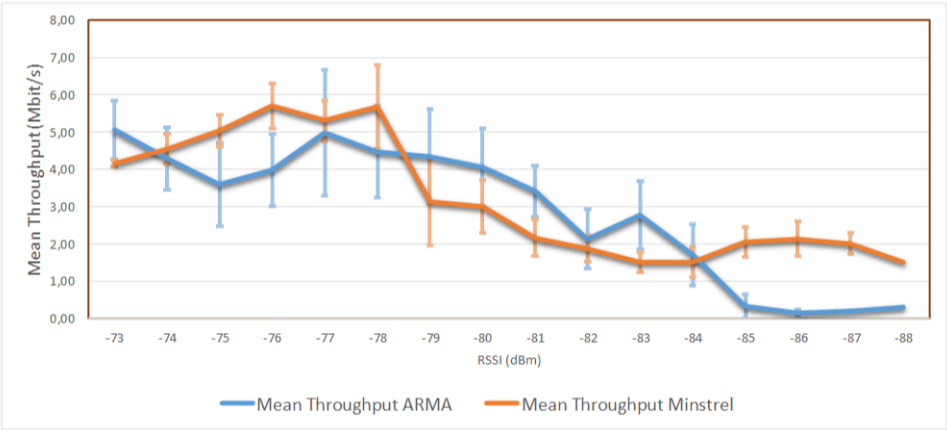


Figure 5.8: Graphic comparing the mean throughput variation with RSSI for ARMA and Minstrel at REX15.

As we see in Figure 5.6 we had an improvement exceeding 80% but for very bad signal strength Minstrel has a bigger mean throughput.

Table 5.6: Mean throughput gain comparing ARMA to Minstrel in REX15

RSSI (dBm)	Mean Gain %
-72	**4,42**
-73	**21,42**
-74	-5,61
-75	-28,66
-76	-30,23
-77	-6,03
-78	-21,29
-79	**38,64**
-80	**34,96**
-81	**58,47**
-82	**14,78**
-83	**84,02**
-84	**13,25**
-85	-84,67
-86	-93,85
-87	-90,00

These differences from what we expected from laboratory experiments can be justified by the number of samples that is relatively lower. Additionally, being almost a maritime environment, all the climate, and the surroundings differ from the controlled almost unaltered laboratory state, which can also contributed to this difference. In addition, the boat movements provoked by the currents may have an influence in this divergences from what we expected.

5.4 MARBED

MARBED is the INESC TEC testbed for maritime communications in a realistic environment. It is currently composed of two nodes: one fixed land node situated in "Edifício Transparente" (Figure 5.11), a commercial building near the shore in Matosinhos (Figure 5.9), the other sea node is put up in the fishing boat mast (Figure 5.10). This node is powered by the boat so it is only on when the boat is operating. The hardware used is equivalent to the one presented in Subsection 4.2. The OS presently running on the Alix board is an older stable version of OpenWRT, namely, OpenWrt Barrier Breaker 14.07.

This was the closest test to a real scenario which we had no control to study the algorithms behaviour. Unfortunately, due to the reduction of the fishing quota, the fishing season was shortened giving us less time to collect data.

Figure 5.9: MARBED test local.

Figure 5.10: Fishing ship with material on board.

Figure 5.11: MARBED land node.

5.4.1 MARBED Evaluation Setup

Since we only had physical access to the sea node when it needed maintenance, we bring up another node (Figure 5.12), to which we had access, to communicate with the node present on the fishing boat. We did this in order not to bother the fisherman during their daily work.

Figure 5.12: Intermediate node montage to access the sea node.

The MARBED network topology can be represented as follows in Figure 5.13. Our extra node had access to the VLAN2 in this case. The node placed on land had connection to the Internet and we could remotely access it.

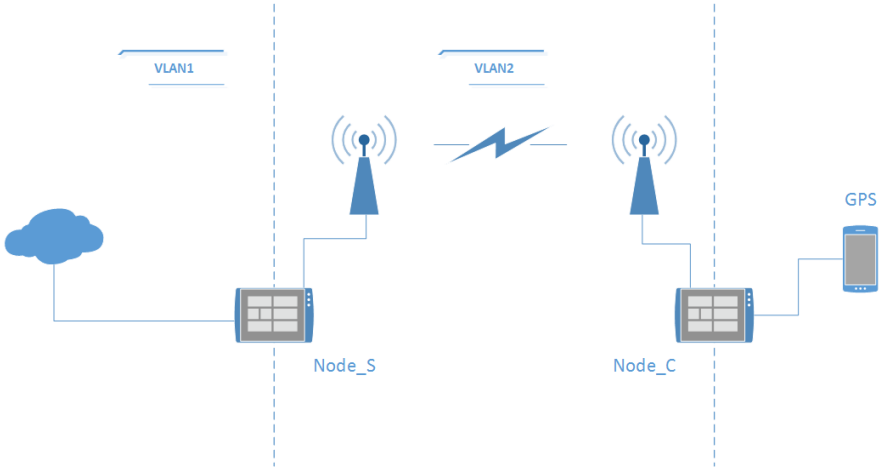


Figure 5.13: MARBED tests network topology.

5.4.2 Evaluation Results

For this test set, it was difficult to collect reliable data: we had little time with a successful link. For this to happen, the boat has to be in range of the antenna and required a clear line of sight. During the time we run tests, the few times the link was created the connection was very fragile. Still we were able to retrieve some data showed in tables 5.7 and 5.8. Overall we still obtained higher mean throughput with ARMA with lower signal strength. In compliance with the other tests, for lower RSSI, ARMA still have higher mean throughput, in most cases.

Table 5.7: Collected data synopsis for ARMA at MARBED.

RSSI (dBm)	#N	Mean (Mbit/s)	StDv
-80	9	5,08	4,39
-81	11	3,07	2,75
-82	7	2,02	1,75
-83	3	5,04	1,33
-84	7	1,50	2,05
-85	8	3,71	4,21
-86	10	2,73	2,92
-87	21	4,65	3,66
-88	21	3,83	3,39
-89	25	2,75	3,22
-90	12	5,05	3,61
-91	32	1,63	1,54

Table 5.8: Collected data synopsis for Minstrel at MARBED.

RSSI (dBm)	#N	Mean (Mbit/s)	StDv
-80	5	1,08	1,42
-81	10	3,93	3,69
-82	16	2,80	2,72
-83	13	3,07	4,09
-84	7	1,21	1,84
-85	8	3,86	4,47
-86	7	2,60	2,50
-87	20	3,16	3,67
-88	12	3,63	3,76
-89	32	2,37	2,58
-90	33	2,67	2,44
-91	3	4,01	2,58

The results obtained were in line of the others performed even though instability of the link probably may caused some errors to occur. Yet, as showed in Figure 5.14 and Table 5.9, considering the same conditions of random errors for both algorithms, ARMA still performed above the Minstrel mean for several RSSI values. We can also see the effect of link instability in the standard deviation values (Table 5.9) and the 95% confidence intervals (Figure 5.14), that are much higher than the other experiments. Even though there were not many results and not very conclusive, we testify the rough environment and the problems that it may cause and we still know the connections can be a reality.

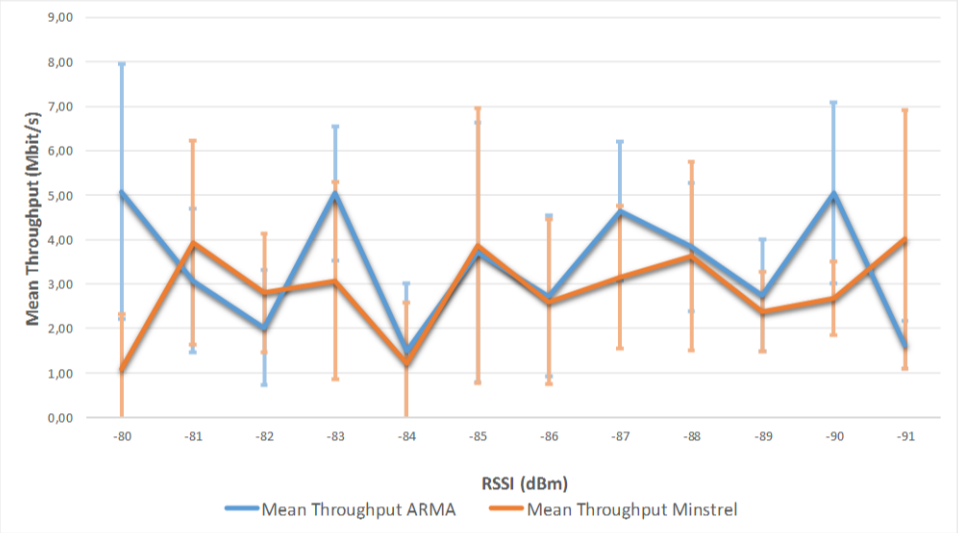


Figure 5.14: Graphic comparing the mean throughput variation with RSSI for ARMA and Minstrel at MARBED.

Table 5.9: Mean throughput gain comparing ARMA to Minstrel in MARBED tests

RSSI (dBm)	**Mean Gain %**
-80	**368,47**
-81	-21,79
-82	-27,93
-83	**63,96**
-84	**23,27**
-85	-3,99
-86	**4,71**
-87	**47,07**
-88	**5,48**
-89	**16,08**
-90	**89,20**
-91	-59,26

5.5 Summary and Discussion

We presented in this chapter three distinct case scenarios where we tested our Adaptive Maximum Transmission Unit and Rate Algorithm and compared it against the state of the art and most used algorithm, Minstrel. We gradually change the test cases from a more stable and regulated conditions to a more realistic scenario in a maritime environment but still controlled and finally a more extreme environment where we had no command. The tests consisted in time intervals running each algorithm at a time and capture the RSSI values and record the corresponding throughput. In the end we compared the mean values of throughput for each value of RSSI. We obtained better results (up to 84% gain), in the REX15 test than in the laboratory test (40% gain). This could be because the laboratory test scenario was too idle and the received signal was not so bad as observed in the real environment. We validated our algorithm in the experimental tests we conducted and concluded that it outperforms its counterpart in a situation where the communication link is poor and inconstant, like in maritime environments. Having also validated the RSSI prediction algorithm, from previous researches, was a plus and endorsed our work.

Our results lead us to conclude that a combined MTU and Rate adaptation to the channel state can improve the overall throughput.

Chapter 6

Conclusion and Future Work

This work is a continuation of the efforts in improving maritime wireless communications and aimed at adapting IEEE 802.11 to the specific maritime characteristic. We wanted to know if adapting simultaneously the MTU and data rate would had a positive impact in achieving higher throughput.

With our work we determine that Wi-Fi is a viable solution to provide wireless broadband communication in maritime environment. The results obtained lead us to conclude that changing dynamically both MTU and rate enhance the communication throughput when the signal strength is weak. The relation between the MTU and rate we introduced was confirmed and the ideal pair chosen by our algorithm proved to improve the throughput achieved. In our tests we observed realistic gains of over 80% mean throughput using UDP. Even though we do not consider the results of MARBED experiment conclusive they show some room for improvement. ARMA proves to be adapted to a maritime environment, despite all its singular characteristics. With this dissertation we were also able to validate the RSSI prediction algorithm from previous work.

With the conclusion of this work we have some suggestions for future research work:

- **Use different technologies** –The use of ARMA together with IEEE 802.11n if the opportunity presents itself once it introduces new tools, like MIMO, that would be interesting to analyse in maritime environment;
- **Congregate with Radiotap**[1] – Instead of using a RSSI prediction use information available in radiotap headers. Having a packet to packet information can have a great impact in order to select the optimum rate and MTU;
- **Considering Transmission Power and Bandwidth** – Consider in addition to MTU and rate, dynamically adjust the transmission power and bandwidth combined and how it affects the communication;
- **Hybrid Algorithm** – Instead of using only the channel state information, include other type of information, such as packet loss;

[1] http://www.radiotap.org/

- **Use other frequencies** – Compare the obtained results in 5.8 GHz with other frequency band such as 700 MHz (TV white spaces).

References

[1] Olivia Swift. Seafarers' access to wifi and wimax in ports, 2013.

[2] Fritz Bekkadal. Emerging Maritime Communications Technologies. *9th International Conference on Intelligent Transport Systems Telecommunications*, pages 358–363, 2009.

[3] João Miguel de Oliveira Magalhães. MAC forWi-Fi-based Land-Sea Communications. *MSc thesis, Faculdade de Engenharia, Universidade do Porto*, 2014.

[4] Mário Jorge de Matos Lopes. Comunicações marítimas Wi-Fi usando a banda 5.8 GHz. *MSc thesis, Instituto Superior de Engenharia do Porto*, 2013.

[5] YoungBum Kim, JongHun Kim, YuPeng Wang, KyungHi Chang, JongWon Park, and YongKon Lim. Application scenarios of nautical ad-hoc network for maritime communications. In *OCEANS 2009, MTS/IEEE Biloxi-Marine Technology for Our Future: Global and Local Challenges*, pages 1–4. IEEE, 2009.

[6] Mohamed Manoufali, Hamada Alshaer, Peng Yong Kong, and Shihab Jimaa. An overview of maritime wireless mesh communication technologies and protocols. *IJBDCN*, 10(1):1–29, 2014.

[7] Ming-TuoZhou et al. Triton: high-speed maritime wireless mesh network. *Wireless Communications, IEEE*, 20(5):134–142, 2013.

[8] Khurram Shabih Zaidi, Varun Jeoti, and Azlan Awang. Wireless backhaul for broadband communication over sea. In *IEEE Malaysia International Conference on Communications (MICC)*, pages 298–303. IEEE, 2013.

[9] Ali Nawaz Naqvi, Ash Mohammad Abbas, and Tofik Ali Chouhan. Performance evaluation of fixed and mobile wimax networks for udp traffic. *International Journal of Advanced Research in Computer and Communication Engineering*, 2012.

[10] Ming-Tuo Zhou, Hiroshi Harada, Peng-Yong Kong, and JS Pathmasuntharama. Wireless mesh networking for maritime intelligent transportation systems, 2010.

[11] Ha Cheol Lee. DCF Throughput Analysis of IEEE 802.11a/g/n-based Mobile LAN over Correlated Fading Channel, 2011.

[12] Luciano Jorge and Silva Santos. Wi-Fi Maritime Communications using TV White Spaces. *MSc thesis, Faculdade de Engenharia, Universidade do Porto*, 2013.

[13] Nishat Kamran et al. On the effectiveness of high-speed wlan standards for long distance communication. In *IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pages 145–146. IEEE, 2014.

[14] Ioannis Pefkianakis, Yun Hu, Starsky HY Wong, Hao Yang, and Songwu Lu. Mimo rate adaptation in 802.11 n wireless networks. In *Proceedings of the sixteenth annual international conference on Mobile computing and networking*, pages 257–268. ACM, 2010.

[15] Tor Martin Slåen. Classifying rate adaptation algorithms in ieee 802.11 b/g/n wireless networks, 2012.

[16] Meraki, 2011. Accessed: 13/12/14. URL: https://meraki.cisco.com/lib/pdf/meraki_whitepaper_802_11n.pdf.

[17] Yuxia Lin and Vincent WS Wong. Wsn01-1: frame aggregation and optimal frame size adaptation for ieee 802.11 n wlans. In *IEEE Global telecommunications conference. GLOBECOM'06.*, pages 1–6. IEEE, 2006.

[18] Stratos Keranidis et al. Experimental evaluation and comparative study on energy efficiency of the evolving ieee 802.11 standards. In *Proceedings of the 5th international conference on Future energy systems*, pages 109–119. ACM, 2014.

[19] Saad Biaz and Shaoen Wu. Rate adaptation algorithms for ieee 802.11 networks: A survey and comparison. In *Computers and Communications, 2008. ISCC 2008. IEEE Symposium on*, pages 130–136. IEEE, 2008.

[20] M A Mohamed, W B Bahget, and S S Mohamed. A performance evaluation for rate adaptation algorithms in ieee 802. 11 wireless networks. *International Journal of Computer Applications*, 99(4):54–59, 2014.

[21] Linux wireless wiki, minstrel. Accessed: 10/02/15. URL: https://wireless.wiki.kernel.org/en/developers/documentation/mac80211/ratecontrol/minstrel.

[22] Anwar Hithnawi. An On-Demand Rate-Adaptation Mechanism For IEEE 802.11 Networks, 2011.

[23] Evaluation of the Minstrel Rate Adaptation Algorithm in IEEE 802.11g WLANs. Accessed: 07/01/15. URL: https://internetnz.nz/sites/default/files/icc_13_final_0.pdf.

[24] Introduction to Wireless LAN Measurements. Accessed: 03/01/15. URL: http://download.ni.com/evaluation/rf/Introduction_to_WLAN_Testing.pdf.

[25] Andrea Goldsmith. Wireless Communications, 2005. Accessed: 09/02/15. URL: http://web.cs.ucdavis.edu/~liu/289I/Material/book-goldsmith.pdf.

[26] John G Proakis. *Intersymbol Interference in Digital Communication Systems*. Wiley Online Library, 2001.

[27] Marvin K Simon and Mohamed-Slim Alouini. A unified approach to the performance analysis of digital communication over generalized fading channels. *Proceedings of the IEEE*, 86(9):1860–1877, 1998.

[28] Hanwu Wang, Weijia Jia, Geyong Min, and Ieee. Effective Channel Exploitation in IEEE 802.16j Networks for Maritime Communications. *31st International Conference on Distributed Computing Systems (Icdcs 2011)*, 2011.

[29] Chee-Wei Ang and Su Wen. Signal strength sensitivity and its effects on routing in maritime wireless networks. In *Local Computer Networks, 2008. LCN 2008. 33rd IEEE Conference on*, pages 192–199. IEEE, IEEE Computer Society, 2008.

[30] Rosario Giuseppe Garroppo, Stefano Giordano, and Davide Iacono. Experimental and simulation study of a wimax system in the sea port scenario. In *IEEE International Conference on Communications, 2009. ICC'09.*, pages 1–5. IEEE, 2009.

[31] Mingyi Zhu, Xiangyang Zhao, and Yu Zhang. Study on a Sea Radio-Wave Propagation Loss Model, April 2010.

[32] Sébastien Thon and Djamchid Ghazanfarpour. Ocean waves synthesis and animation using real world information. *Computers & Graphics*, 26(1):99–108, 2002.

[33] Cycloids vs trochoids. Accessed: 10/01/15. URL: http://curvebank.calstatela.edu/cycloidmaple/Oids/OidEG1.html#Cyc/Troc.